

UNIVERSIDAD CARLOS III DE MADRID

ESCUELA POLITÉCNICA SUPERIOR

**GRADO EN INGENIERÍA EN TECNOLOGÍAS DE
TELECOMUNICACIÓN**



TRABAJO FIN DE GRADO

**TRATAMIENTO DE IMAGENES
DERMATOSCÓPICAS PARA AYUDA AL
DIAGNÓSTICO Y SEGUIMIENTO DE LA
ALOPECIA**

AUTOR: Alejandro Blanco Pizarro

TUTOR: Fernando Díaz de María

Septiembre de 2015

Agradecimientos

Se ha acado un ciclo de mi vida de 5 años con grandes experiencias y emociones. Me han parecido unos hermosos años, en los cuales he disfrutado mucho con mis compañeros de clase y he aprendido a valorar el trabajo personal y el de los demás.

Lo primero es dar gracias a mi familia, por todo el apoyo recibido para seguir estudiando y sobretodo a mi madre por ayudarme en momentos difíciles de estos años.

También quiero dar gracias a mi tutor Fernando, por su dedicación y confianza que ha tenido conmigo y al proyecto. Y a javier por su gran aportación a una parte del proyecto.

Cada momento de la vida nos enseña a ser mejor persona, aprendamos de ella y vivamos cada época con la mayor esperanza y confianza posible.

Resumen

La alopecia es una enfermedad que afecta a gran parte de la población, 50 % de los hombres y el 15 % de las mujeres. De acuerdo a muchos estudios, la caída del cabello causa un impacto emocional muy fuerte, perdiendo el individuo la confianza y disminuyendo su autoestima.

Hay diferentes tipos de alopecia, cada una de ellas tiene un tratamiento diferente, lo importante es que el especialista valore el estado del paciente, determine la causa y establezca un tratamiento apropiado.

Los dermatólogos necesitan una herramienta de apoyo, que determine ,la mejoría o empeoramiento del tratamiento, y cambiarlo si es necesario para no empeorar el estado del paciente.

El propósito del proyecto es proporcionar una herramienta sencilla para el diagnóstico y tratamiento de la alopecia, donde el doctor tenga en todo momento imágenes precisas de la evolución del cabello.

Para cumplir estos objetivos, se hará uso de imágenes dermatoscópicas, son imágenes obtenidas mediante la técnica tricoscopia. La idea de esta técnica es ampliar la zona afectada que se quiere analizar, obteniendo en la foto un aumento de 20 a 400 el cuerpo del pelo.

Para llevar a cabo los siguientes pasos:

- Segmentación de pelos en una imagen.
- Extracción de características del pelo: Color, anchura y orientación.

En la actualidad hay diversos softwares que cubren total o parcialmente los objetivos del proyecto, se puede distinguir:

- DullRazor: Es un software de borrado de pelos programado en C. La parte de interés para el proyecto es el método de segmentación de pelos

que utiliza. Se basa en operaciones morfológicas en escala de grises, para localizar los píxeles de pelos oscuros. Para mejorar la segmentación se usa un filtro de mediana adaptativo.

- TrichoScan: Es un software de estudio del cabello que mide parámetros biológicos, como son la anchura, la densidad y fases de crecimiento del pelo. Para ello se selecciona una zona del cabello de 2mm y se rapa; a los pocos días se procesa la zona y se obtienen los resultados.
- FotoFinder: Software que estudia también parámetros del cabello como anchura, densidad y medición de folículos capilares. Se usa una zona de 5 mm para el estudio del cabello.

Durante el proyecto se ha hecho uso de herramientas básicas de segmentación, como:

- Método de Otsu: Se basa en la segmentación por nivel de gris de la imagen. El propósito de este procedimiento es elegir un umbral de nivel de gris, por el cuál todos los píxeles queden etiquetados como unos o ceros, o visto de otra forma, establecer 2 clases de píxeles.

El potencial del método de Otsu reside en la elección del umbral, es aquel que maximiza la varianza interclase.

- Segmentación basada en derivadas: La derivada de una función indica cambios notables para el estudio, como son máximos, mínimos y puntos de inflexión. La idea de estos métodos es hacer uso de la derivación para localizar cambios en la imagen.

- Primera derivada: Aquí se obtiene el gradiente de la imagen con máscaras discretas(Prewitt) o convolucionando la imagen con filtros derivativos gaussianos, que estiman el gradiente. Ya calcula-

do el gradiente se obtiene el módulo y se localizan los máximos o mínimos locales, que indican una frontera.

Otro método es el Detector de Borde de Canny, se escogen los máximos locales del módulo del gradiente, y un seguimiento para conseguir bordes de la imagen de un solo píxel. Para descartar ruido que aún puede haber, se usan 2 umbrales, fuerte y débil. Solo se unirán los píxeles que hayan superado el umbral débil a los píxeles que hayan superado el umbral fuerte.

- Segunda derivada: Aquí se maneja el operador laplaciano de la imagen, mediante el filtrado de la imagen con una máscara discreta, que estime este operador o convolucionando la imagen con el filtro laplaciano gaussiano. La idea es buscar los cruces por cero, ya que indicarían la presencia de una frontera en ese píxel.

- Morfología matemática: Es una técnica de análisis y tratamientos de estructuras geométricas, para el estudio de la forma, conectividad y tamaño. La información obtenida como resultado de estas técnicas, depende del elemento estructurante que se use. Aquí solo nos centramos en morfología binaria:

Las operaciones morfológicas binarias vistas desde una perspectiva geométrica, dan información estructural de los objetos en la imagen, identificando la zona donde el elemento estructurante quepa parcial o totalmente.

Hay 2 tipos de operadores básicos: erosión y dilatación. La operación de erosión marca la zona de la imagen donde el elemento estructurante quepa totalmente y la operación de dilatación marca la zona donde el elemento quepa parcial o totalmente. También hay 2 operaciones secundarias: apertura y cierre, estas operaciones son combinación de

los operadores básicos.

- Apertura de área: Este operador borra pequeños objetos de una imagen binaria, si el área es menor a un parámetro λ de entrada. Muy útil para eliminar ruido.

El diseño técnico del sistema se basa en 3 partes principales: preprocesado, segmentación y extracción de características:

- Preprocesado: La imagen de entrada en el espacio de color RGB, se procesa para obtener una imagen en escala de grises y después se realiza un filtrado de mediana para eliminar ruido y realzar los bordes.
- Segmentación: Se han usado 3 métodos de segmentación para después realizar una evaluación y escoger la mejor solución. Los métodos usados son:
 - Método de Otsu: Para mejorar la calidad de segmentación del método de Otsu se calcula el umbral, a nivel local y no a nivel general de la imagen. Se particiona la imagen en bloques más pequeños y se calcula el umbral de Otsu en estos bloques. Así se consigue un umbral más preciso.
 - Detector de bordes de Canny: Es importante seleccionar bien los umbrales que usa Canny para detectar los bordes. En cada imagen, se estiman los umbrales en función del módulo del gradiente. Para rellenar los bordes de los pelos se realiza la operación morfológica de cierre, con elemento estructurante de tipo línea en diferentes direcciones, para realizar un rellenado completo.
 - Filtros derivativos Gaussianos: La salida del filtrado derivativo gaussiano está definida por un máximo, al principio de la frontera

del pelo y por un mínimo al final. Gracias a este hecho se consigue obtener una respuesta entre máximo y mínimo.

- Extracción de características: En este punto se estima la orientación y el ancho, y además una descripción del color de los pelos segmentados.

La orientación y el ancho se estima mediante el uso de filtros derivativos gaussianos normalizados, determinando qué filtro, en función de su desviación típica, corresponde con la máxima salida del filtrado.

La descripción del color se representa mediante un histograma, que permite reconocer visualmente el color predominante y el número de píxeles asociados a cada color.

La finalidad del proyecto es determinar la mejor solución al problema. Para ello se realizan pruebas para evaluar la calidad de los diferentes métodos. Al final de la memoria se exponen los resultados y establece la mejor solución.

Viendo los resultados, se escogió el método de Otsu como solución final por sus buenos resultados en la evaluación. La falta de imágenes en el proyecto han hecho no disponer de imágenes de test y solo de entrenamiento, pero los resultados son muy fiables y no habrá ningún problema en usar el sistema en diferentes imágenes.

El fin ideal de este estudio de viabilidad, es llevar a cabo la implantación de un producto comercial. Con todo lo que supone, autorizaciones de pacientes, aumentar las imágenes de base de datos, dando robustez al sistema propuesto en este estudios, búsqueda de financiación y clientes.

Palabras clave: segmentación, Otsu, Canny, morfología, pelo, cabello, orientación, filtro, derivada, gaussiano.

Summary

Alopecia is a pathology that affects in a high part of the population, 50 % of men and 15 % of women. According to many social studies, hair lost causes a psychological impact in the patient, losing his confidence and decreasing his self-esteem.

There are many types of alopecia, each one has a different treatment. The most important thing is to assure the the patient's condition, determining the cause and establishing the appropriate treatment.

Dermatologist need a support tool to determine the improvement or deterioration of the treatment, and if it is necessary to change the treatment in order to not worsen the patient's condition

The project's purpose is to supply an easy tool for the diagnosis and study of alopecia, so that the dermatologist has at all time images of the hair's evolution.

To carry out this objectives, dermathoscopy images are used along the project. This images are taken by trichoscopy method. The idea of this method is to get an enlarge picture of the affected area increased from 20x to 400x.

To do so, we implement the following steps:

- Hair segmentation in an image.
- Property extraction of hair: Color, width and orientation.

Nowadays there are several software utilities which cover fully or partially the project's objectives. We could distinguish:

- DullRazor: Is a hair-erasing software. The interesting part for the project is the method of hair segmentation, which is based on grayscale

closing morphological operation to locate the pixels of dark hair. To improve the segmentation mask, the software uses an adaptive median filter.

- TrichoScan: Is a study hair software which measure the biology's parameters such as width, density and growth phases. To carry out the study, the hair must be shaved at most 2 mm and then dyed. In a few days, the software process the area and get the results.
- FotoFinder: Is a hair study software similar to TrichoScan.

Basic segmentation tools are used during this project, such as:

- Otsu's method: This method is based on the segmentation by gray level. The idea is to choose a gray level threshold and label all pixels with 0 and 1 and set two class of pixels.

The potential of Otsu's method reside in the threshold choice, which maximizes the interclass variance.

- Segmentations based on derives: Derivative of a function show the location of notables changes like maximum, minimum or inflection point. The purpose of these methods is to use derivation to locate alteration in the images.

- First derivative: The gradient is filtered using a discrete mask, or convolving the image with derivative gaussian filter. With the calculated gradient, just need to get the modulus and locate the local maximums or minimums which suggest a edge.

Another method is the Canny border detector which calculates the gradient modulus to select local maximums and a get borders of one pixel width. Canny uses 2 threshold, high and low to

remove noise. Only pixels which surpasses the low threshold will join to the pixels which surpasses the high threshold.

- Second derivative: In this case, the laplacian operator of the image is calculated by discrete mask or convolving with a laplacian gaussian filter. The idea is to find out zero crossings to locate a border in this pixel.

- Mathematic morphology: It is an analysis and geometric structure processing technique, whose objective is the study of its shape, size and connectivity. The results of this technique may vary depending on the used structuring element. In this lecture, we are going to focus on binary morphology:

From a geometric perspective, binary morphologic operations provide structural information of the objects in a given picture, identifying the area where the structuring element fits fully or partially.

There are two types of basic operators: erosion and dilatation. The erosion operation marks the area where the structuring element will fit completely within an image. On the other hand, dilatation operation marks the area where the element fits fully or partially. Two secondary operators can also be found: open and close. These are combinations of the basic operators.

- Area opening: This operator removes small objects in a binary object, only if the area is smaller than a given λ parameter. It is very useful for noise reduction

System's technical design is based on three key components: properties preprocessing, segmentation and extraction:

- Preprocessing: Within the RGB color interval, the input picture is being processed to get a greyscale image and then apply a median filter in order remove noise and border enhance.
- Segmentation: We have been used three segmentation methods so then can perform a test and choose the best solution. Those methods were:
 - Otsu’s method: In order to improve Otsu’s method segmentation quality, threshold is calculated locally instead of picture’s level. The picture is divided in small blocks and then the Otsu’s threshold is calculated for each block. That is the way to get more precise thresholds.
 - Canny’s border detection: Threshold selection is very important for Canny’s border detection. Threshold estimation of every picture is based on gradient modulus. To perform an full filling, the close morphologic operator is used, having lines in different directions as a structuring element.
 - Gaussian derivative filters: The output of the gaussian derivative filter y defined by a maximum, the beginning of the hair border, and a minimum, the end of the border. Due to this fact, it is possible to get a solution between maximum and minimum.
- Properties extraction: At this point, width and orientation of the segmented hair is estimated, plus a color description.

Width and orientation is estimated using normalized Gaussian derivative filters, determining which filter matches with the maximum filtering output, based on its standard deviation.

Color description is represented by an histogram, which allows to visually recognize the dominant color and pixels number of every color.

The aim of this project is to establish the best solution to the problem. To achieve it, method's quality testing have been performed. At the end of this report, all results are shown and the best solution is chosen.

Having the results, the best solution was Otsu's method, because of its excellent output values during the test. The lack of pictures along this project have made to have only training pictures and none for testing. However, the results are reliable anyway, and there won't be any problem in using some other pictures in the system.

The final aim of the viability study is to successfully develop a product for commercial use. It includes, patient's authorizations, more pictures in database to make the system stronger, fundraising and customers search.

Palabras clave: segmentation, Otsu, Canny, morfology, hair, orientation, filter, derivative, gaussian.

Índice general

Lista de figuras	xix
1 Introduction	1
1.1 Motivation	1
1.2 Objectives	3
1.3 Strucutre of the report	4
1.4 Regularity framework	4
2 Herramientas básicas de segmentación	6
2.1 Método de Otsu	6
2.1.1 Elección de umbral	6
2.2 Segmentación basada en derivadas	8
2.2.1 Primera derivada	8
2.2.1.1 Derivative of Gaussian(DoG)	10
2.2.1.2 Detector de bordes de Canny	13
2.2.2 Segunda derivada	15
2.2.2.1 Laplacian of Gaussian (LoG)	15
2.3 Morfología matemática	17
2.3.1 Operaciones morfológicas binarias	18
2.3.1.1 Erosión	18
2.3.1.2 Dilatación	20

2.3.1.3	Apertura y cierre	22
2.3.2	Operaciones morfológicas en escala de grises	23
2.3.2.1	Erosión	24
2.3.2.2	Dilatación	24
2.3.2.3	Apertura y cierre	25
2.3.2.4	Transformación “top-hat” y “bottom-hat”	27
2.4	Apertura de área	30
3	Estado del arte	32
3.1	Dull Razor	32
3.2	TrichoScan	36
3.3	FotoFinder Trichoscale	37
3.4	Segmentación de los vasos sanguíneos de la retina	37
4	Diseño técnico	42
4.1	Preprocesado	43
4.1.1	Componentes de colores	43
4.1.2	Filtros estadísticos de ordenamiento	45
4.2	Segmentación	50
4.2.1	Método de Otsu	50
4.2.1.1	Factor de calidad	52
4.2.1.2	Acotar los umbrales de gris	54
4.2.1.3	Suavizado de umbral	55
4.2.2	Detector de borde de Canny	56
4.2.3	Derivative of Gaussian (DoG)	59
4.2.4	Borrado de pequeños objetos	63
4.3	Etiquetado	65
4.4	Extracción de características	68

4.4.1	Estimación de la orientación	68
4.4.2	Estimación del ancho	69
4.4.2.1	Basado en filtros DoG normalizados	69
4.4.2.2	Basado en la respuesta del filtro DoG	71
4.4.3	Estimación del color	73
4.4.3.1	Espacios de color	73
4.4.4	Descripción del color	76
5	Evaluación	78
5.1	Máscaras de referencia	78
5.2	Medidas de evaluación	78
5.3	Resultados	79
6	Planificación y presupuesto del proyecto	82
6.1	Planificación	82
6.2	Presupuesto	83
7	Conclusion and futures lines	86
7.1	Conclusions	86
7.2	Futures lines	86
	Anexos	88
A	Relación σ con ancho	89
B	Imágenes usadas en el proyecto	91
C	Imágenes usadas para la relación de σ con el ancho en px	96
D	Imagen sintética para la evaluación de la estimación del ancho	111

Índice de figuras

1.1	Dermatoscopy image	3
2.1	(a) Imagen original de ejemplo, (b) Gradiente en horizontal y (c) Gradiente en vertical	9
2.2	(a) Módulo del gradiente, (b) Aplicando umbral	10
2.3	Filtro gaussiano de tamaño 10x10 con desviación típica igual a 2, normalizado y generado por matlab	10
2.4	(a) DoG orientacion x, (b) DoG orientacion y, (c) DoG orien- tacion 45 y (d) DoG orientacion 135 grados	11
2.5	(a) Ejemplo(pequeños pulsos), (b) Filtro DoG,(c) Resultado aplicando convolucion con DoG	12
2.6	Filtro LoG con sigma 2 y tamaño 10x10 generado con matlab	16
2.7	(a) Imagen en escala grises, (b) Resultado de aplicar filtrado .	16
2.8	Detección de bordes usando LoG	16
2.9	Ejemplo de operación de erosión	19
2.10	Operación de erosión binaria (a) Imagen original, (b) Resul- tado de aplicar erosión	20
2.11	Ejemplo de operación de dilatación	21
2.12	Operación de dilatación binaria (a) Imagen original, (b) Re- sultado de aplicar dilatación	21

2.13	Apertura	22
2.14	Cierre	23
2.15	Ejemplo de reflexión	25
2.16	Ejemplo para los operadores de apertura y cierre en escala de grises, Lenna	26
2.17	(a) Aplicando apertura mediante elemento estructurante ti- po disco de radio 2, (b) Aplicando cierre mediante elemento estructurante tipo disco de radio 2	27
2.18	Imagen con granos de arroz	28
2.19	(a) Aplicando apertura mediante elemento estructurante tipo cuadrado de lado 8, (b) Aplicando “top-hat” mediante ele- mento estructurante cuadrado de lado 8	29
2.20	Imagen con granos de arroz	30
2.21	(a) Aplicando cierre mediante elemento estructurante tipo cuadrado de lado 8, (b) Aplicando “bottom-hat” mediante elemento estructurante uadrado de lado 8	30
2.22	(a) Imagen original, (b) Apertura de área binaria	31
3.1	Elementos estructurantes para la operación de cierre en es- cala de grises (a) Elemento estructurante horizontal centrado en (6,0), (b) Elemento estructurante en diagonal centrado en (4,4), (c) Elemento estructurante vertical centrado en (0,6)	33
3.2	(a) Imagen original de ejemplo, (b) Mascara aplicando Dull- Razor (c) Aplicando interpolación	36
3.3	Ejemplo de imagen de TrichoScan	37
3.4	Ejemplo de vasos sanguíneos de la retina [23]	38
4.1	Flujograma del sistema	42

4.2	Imagen del proyecto en componente CL	43
4.3	Imagen del proyecto en componente CP	44
4.4	Padding con información de los pixeles contiguos	45
4.5	(a) Imagen con ruido, (b) Resultado de aplicar filtro de mediana	47
4.6	Influencia de la iluminación	48
4.7	(a) Ventana 3x3, (b) Ventana 4x4 y (c) Ventana 5x5	49
4.8	(a) Ruido por luz, (b) Filtrado gaussiano sigma 3	50
4.9	(a) Imagen en escala grises, (b) Resultado de aplicar el método de Otsu	51
4.10	(a) Imagen en escala grises, (b) Otsu a nivel local	52
4.11	Factor de calidad CL	53
4.12	Factor de calidad CP	53
4.13	Factor de calidad CL aplicando umbral	54
4.14	Factor de calidad CP aplicando umbral	54
4.15	Umbrales Otsu CL	54
4.16	Umbrales Otsu CP	55
4.17	Umbrales Otsu en CP acotados	56
4.18	Resultado final de aplicar el método de Otsu	56
4.19	(a) Imagen original en CL, (b) Resultado de aplicar Canny .	57
4.20	(a) Aplicando Canny, (b) Aplicando operación de cierre a la salida de Canny	58
4.21	Ejemplo de la utilidad del factor d	59
4.22	Ejemplo de obtención del centro del pelo (a) Ejemplo de filtrado DoG, (b) Resultado de aplicar la obtención del centro a la salida del filtrado DoG	60
4.23	Ejemplo de una cota de nivel de gris para cada píxel (a) Antes, (b) Después	61

4.24	Histograma del resultado de aplicar una cota al nivel de gris(a) Resultado al que pertenece el histograma, (b) Histograma . . .	62
4.25	Ejemplo de aplicar una cota en la respuesta del centro en segmentación DoG(a) Antes, (b)Despues	63
4.26	Aplicando apertura de área en DoG(a) Antes, (b)Después . .	64
4.27	Orientación representado por colores. Rojo simboliza 0°, ver- de 45°, azul 90° y amarillo 135°	69
4.28	(a) Imagen original, (b) Estimación del ancho basado en DoG normalizados	71
4.29	Representación de la máquina de estados	72
4.30	(a) Imagen original, (b) Estimación del ancho basado en la respuesta del filtro DoG y máquina de estados	73
4.31	Cubo RGB	74
4.32	HSV	75
4.33	Representación CIELAB	76
4.34	Histograma en función de HUE	77
5.1	Representación HD	79
6.1	83
6.2	83
6.3	(a) Parte 1 del Gantt, (b) Parte 2 del Gantt	83
C.1	Línea con ancho 3 px	96
C.2	Línea con ancho 4 px	97
C.3	Línea con ancho 5 px	97
C.4	Línea con ancho 6 px	98
C.5	Línea con ancho 7 px	98
C.6	Línea con ancho 8 px	99

C.7 Línea con ancho 9 px	99
C.8 Línea con ancho 10 px	100
C.9 Línea con ancho 11 px	100
C.10 Línea con ancho 12 px	101
C.11 Línea con ancho 13 px	101
C.12 Línea con ancho 14 px	102
C.13 Línea con ancho 15 px	102
C.14 Línea con ancho 16 px	103
C.15 Línea con ancho 17 px	103
C.16 Línea con ancho 18 px	104
C.17 Línea con ancho 19 px	104
C.18 Línea con ancho 20 px	105
C.19 Línea con ancho 21 px	105
C.20 Línea con ancho 22 px	106
C.21 Línea con ancho 23 px	106
C.22 Línea con ancho 24 px	107
C.23 Línea con ancho 25 px	107
C.24 Línea con ancho 26 px	108
C.25 Línea con ancho 27 px	108
C.26 Línea con ancho 28 px	109
C.27 Línea con ancho 29 px	109
C.28 Línea con ancho 30 px	110
D.1 Líneas con diferentes anchos y direcciones	111

Chapter 1

Introduction

1.1 Motivation

Alopecia is a pathology that affect in a great part of population, 50 % of men and 15 % of women [1]. According to many social studies the hair lost causes a high emotional impact, the patient lost his confidence, decrease his self-esteem and create preoccupation, anxiety and anger. As you can see it is a social problem that can generate serious psychological problems with a strong feedback.

It should be added the causes of alopecia are very different [2] and those depend on the person, we can distinguish between others:

- Androgenic alopecia, also known as "male pattern baldness", can strike younger as well as older people. In spite of its masculine name, women can get this condition, too. It's genetic, so having a family history can predict if you might inherit it. In both men and women, it's linked to having an excess of male hormones (androgens) around the hair follicles, which can block hair growth. Women are more likely to develop androgenic alopecia after menopause, when they have fewer female hormones.
- Alopecia areata typically causes a few temporary bald patches on the scalp. It tends to run in families and often strikes in childhood. The hair loss seems to be part of an immune system problem, in which the body's natural defences mistakenly attack its own tissue. Once the hair has fallen out in certain spots, new growth is suppressed for weeks

or months. This type of alopecia sometimes affects people who have other autoimmune diseases like thyroid disease, lupus, or pernicious anemia. Sometimes, it may produce complete scalp baldness (alopecia totalis) or total loss of body hair (alopecia universalis).

The important thing is to assess the condition of the patient, see what is the causes and apply the best treatment. Dermatologist have an important rol because the are responsable of applying all of this things. More an more desmatologists demand an analytics tool to estimate the improvement or deterioration of the patient and minimize the psychological impact of the patient

This proyect proposes a solution of this problem through hair segmentation tool and estimating characteristics of the hair as colour, width, and quantity. The most important is to oversee the treatment, compare results and estimate the condition of the patient.

1.2 Objectives

The project objective is to create a analytic tool easy to use for alopecia diagnostic.

Dermatoscopic images have been ceded by medical clinic in Madrid to improvement the project. This images are taken by trichoscopy method [28]. The idea of this method is to get an enlarge picture of the affected area increased from 20x to 400x.



Figura 1.1: Dermatoscopy image

The problem is getting parameter of hair and compare those in a differentes condition of the treatment. Knowing the improvement or deterioration of the treatment:

To do so, we implement the following steps:

- Hair segmentation: Many segmentation process will explain during the project, seeing the features of each one and choosing the best segmentation process.
- Property extraction of hair:

- Color
- Width
- Orientation

1.3 Strucutre of the report

The project's report is divided by:

- Chapter 1: Introduction, motivation, project's objective, structure of the report and regularity framework.
- Chapter 2: Basic segmentation tools have been introduced to understand the state of the art.
- Chapter 3: State of the art.
- Chapter 4: The technical solutions.
- Chapter 5: The results of the technical solution have being exposed in this chapter.
- Chapter 6: Planification y budget have being exposed in this chapter.
- Chapter 7: Conclusion and futures lines.

1.4 Regularity framework

Dermatoscopic images have been ceded by medical clinic in Madrid following the Law Organic of date protection [30] of personal position, known as LOPD, law nº 15/1999, de 13 december, article number 18.4 of Spanish Constitution, "... limit the use of information technology to ensure the honor, personal and family privacy of citizens and the legitimate exercise of their rights".

This project is only a viability study, if the project will take place, we would need:

- Patient's authorizations.
- Approval of the Ethics Committee of Hospital partner to carry out the investigation.

Capítulo 2

Herramientas básicas de segmentación

En este capítulo se expone las herramientas básicas de segmentación:

2.1 Método de Otsu

Otsu [7] se basa en la umbralización por nivel de gris creando 2 clases, una clase donde se encontrarían todos los píxeles por debajo del umbral de Otsu y la segunda clase por encima.

Al aplicar un umbral, T , la imagen quedará binarizada atendiendo a:

$$g(x, y) = \begin{cases} 1 \leftrightarrow f(x, y) < T \\ 0 \leftrightarrow f(x, y) \geq T \end{cases}$$

siendo $f(x, y)$ la representación de la imagen en escala de grises y $g(x, y)$ el resultado de aplicar el umbral. El método de Otsu da el umbral que maximiza la varianza interclase.

2.1.1 Elección de umbral

Una imagen en escala de grises contiene $M \times N$ píxeles cuyos niveles de gris se encuentran entre 0 y $L - 1$, donde L suele ser un valor de 256 utilizando un byte para representar la escala de grises. El número de píxeles con nivel de gris i se denota como f_i y la probabilidad de ocurrencia del nivel de gris i en la imagen viene dada por:

$$p_i = \frac{f_i}{(M \times N)}$$

La probabilidad de cada clase será:

$$w_1(T) = \sum_{i=0}^T p_i \quad w_2(T) = \sum_{i=T+1}^{L-1} p_i$$

Definimos las medias para ambas clases:

$$\mu_1(T) = \sum_{i=0}^T \frac{ip_i}{w_1(T)} \quad \mu_2(T) = \sum_{i=T+1}^{L-1} \frac{ip_i}{w_2(T)}$$

Y la media global

$$\mu_G = \sum_{i=0}^{L-1} ip_i$$

Se puede verificar:

$$w_1(T) + w_2(T) = 1 \quad w_1(T)\mu_1(T) + w_2(T)\mu_2(T) = \mu_G$$

Lo siguiente es definir la varianza de cada clase:

$$\sigma_1^2(T) = \sum_{i=0}^T (i - \mu_1(T))^2 \frac{ip_i}{w_1(T)} \quad \sigma_2^2(T) = \sum_{i=T+1}^{L-1} (i - \mu_2(T))^2 \frac{ip_i}{w_2(T)}$$

Y la varianza global:

$$\sigma_G^2 = \sum_{i=0}^{L-1} (i - \mu_G)^2 p_i$$

Otsu define la varianza entre clases como:

$$\sigma_O^2(T) = w_1(T)(\mu_1(T) - \mu_G)^2 + w_2(T)(\mu_2(T) - \mu_G)^2$$

Para una umbralización de 2 niveles Otsu propone que el umbral óptimo T^* es:

$$T^* = \max_T \sigma_O(T)^2$$

Otsu también provee de un factor de calidad que indica como están de separadas las dos clases:

$$S(T^*) = \frac{\sigma_O(T^*)^2}{\sigma_G^2} \quad 0 < S(T^*) < 1$$

Un valor próximo a uno nos indica que la imagen tiene dos clases bien definidas y separadas y por el contrario un valor próximo a cero indica dos clases juntas y muy semejantes.

2.2 Segmentación basada en derivadas

La operación de derivación es una herramienta útil a la hora de estudiar cambios notables en la función, máximos, mínimos y puntos de inflexión. Si se extrapola este procedimiento en imágenes se puede localizar cambios en la imagen, fronteras entre diferentes objetos para luego poder procesar estos objetos en función de la necesidad. Se distingue segmentación basada en primera y segunda derivada:

2.2.1 Primera derivada

La idea de este procedimiento es hacer uso del gradiente de la imagen ya que indica la dirección de la máxima variación, en módulo y fase. Donde haya un máximo del módulo del gradiente habrá un cambio notable en la imagen.

Para estimar el gradiente en imágenes se usan un tipo de máscaras con las que se filtra la imagen. Los máscaras típicas son:

- Prewitt: Estas máscaras se usan para estimar el gradiente, bien en vertical o bien en horizontal:

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -1 & 0 & 1 \\ -1 & 0 & 1 \end{bmatrix} \quad H_y = \begin{bmatrix} -1 & -1 & -1 \\ 0 & 0 & 0 \\ 1 & 1 & 1 \end{bmatrix}$$

- Sobel: Un poco más complejo que Prewitt, dando mas protagonismo al píxel central:

$$H_x = \begin{bmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{bmatrix} \quad H_y = \begin{bmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{bmatrix}$$

Se calcula el módulo del gradiente en horizontal y vertical filtrando por mediante estas máscaras, obteniendo:

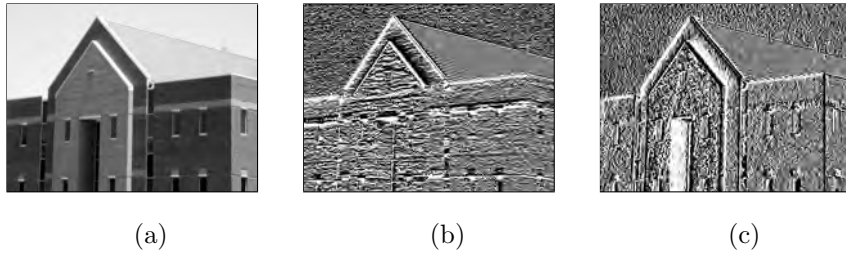


Figura 2.1: (a) Imagen original de ejemplo, (b) Gradiente en horizontal y (c) Gradiente en vertical

Después se realiza la raíz de la suma de los cuadrados de las salidas del filtro en horizontal y vertical para obtener el módulo del gradiente en ambas direcciones. Después se debe aplicar un umbral del cual un 5 % o 10 % de los píxeles superan este umbral, los de más valor. Lo normal es hacer uso del histograma acumulado normalizado y escoger un umbral de modo que el 90 % o 95 % de los píxeles quedan por debajo.

Todo este proceso se llama segmentación por umbral.

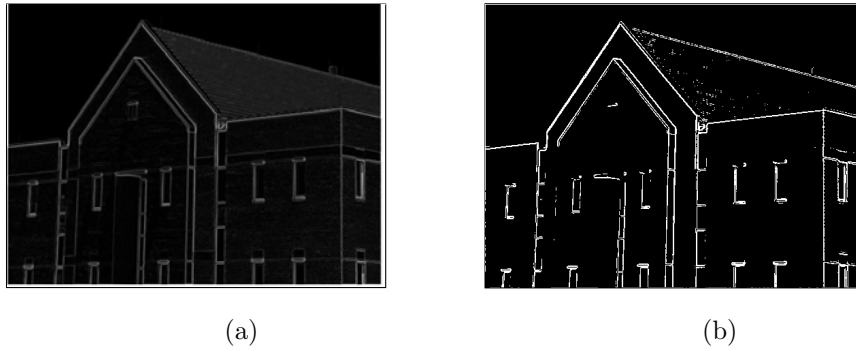


Figura 2.2: (a) Módulo del gradiente, (b) Aplicando umbral

2.2.1.1 Derivative of Gaussian(DoG)

Este tipo de segmentación está basado en el uso de los filtros derivativos Gaussianos en 2D, derivative of Gaussian(DoG). Lo primero es definir la Gaussiana en 2 dimensiones:

$$G(x, y, \sigma) = \frac{1}{2\pi\sigma^2} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

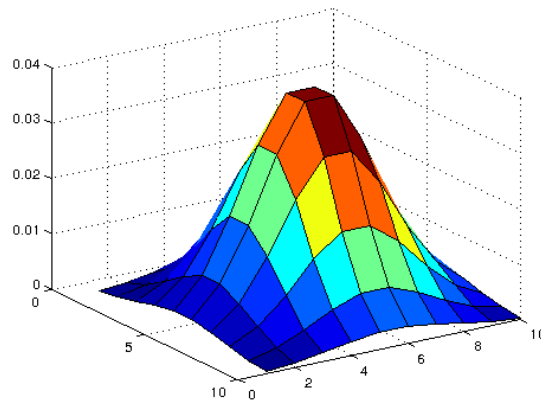


Figura 2.3: Filtro gaussiano de tamaño 10x10 con desviación típica igual a 2, normalizado y generado por matlab

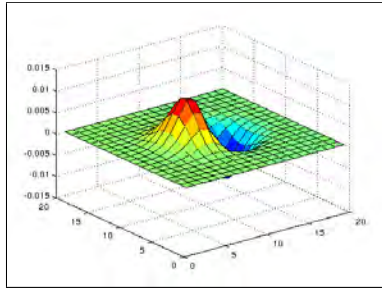
Calculando el gradiente [16] en (x, y) de la Gaussiana:

$$G_x(x, y) = \frac{\partial}{\partial x} G(x, y, \sigma) = \frac{-x}{\sqrt{2\pi}\sigma^3} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

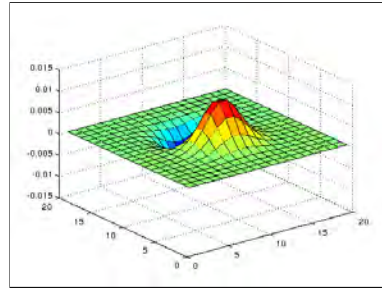
$$G_y(x, y) = \frac{\partial}{\partial y} G(x, y, \sigma) = \frac{-y}{\sqrt{2\pi}\sigma^3} e^{-\frac{(x^2+y^2)}{2\sigma^2}}$$

Estos filtros permiten detectar líneas en cualquier dirección por ser orientables, se pueden rotar para conseguir una versión del filtro original pero rotada en función de un ángulo; para ello se usa una combinación lineal de los filtros básicos:

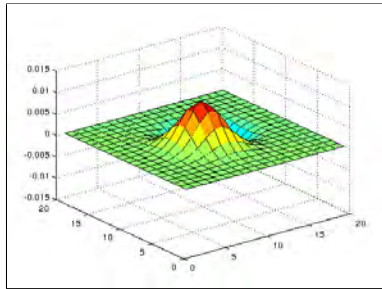
$$G_\theta(x, y) = G_x \cos(\theta) + G_y \sin(\theta)$$



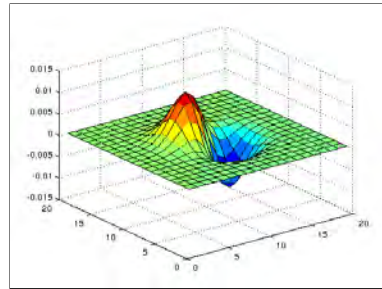
(a)



(b)



(c)



(d)

Figura 2.4: (a) DoG orientacion x, (b) DoG orientacion y, (c) DoG orientacion 45 y (d) DoG orientacion 135 grados

Después se debe calcular la respuesta [15] del filtro usando la convolución:

$$R_i(x, y) = G_i(x, y) * I(x, y)$$

Para entender mejor la respuesta se puede relacionar los picos de la imagen como un tren de delta que se convoluciona con el filtro, obteniendo una versión desplazada de este en la posición del tren de deltas. Se muestra a continuación un ejemplo en una dimensión (en continuo):

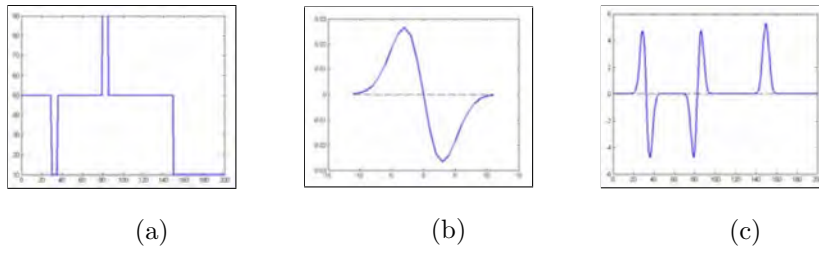


Figura 2.5: (a) Ejemplo(pequeños pulsos), (b) Filtro DoG,(c) Resultado aplicando convolucion con DoG

Se debe añadir que los pulsos en el ejemplo se pueden extrapolar en imágenes a fronteras donde los positivos se asemejan a fronteras de paso a color mas claro y las negativas hacia oscuro.

Aparte se ve que el cruce por cero de la respuesta indica el centro del pulso y deja a los lados un máximo y un mínimo a la misma distancia del centro, d . Además el primer máximo o mínimo indicaría el inicio de la frontera y el segundo el final, aprovechando esta respuesta se obtener el centro gracias a este procedimiento:

$$C(x, y) = \begin{cases} R(x - d, y) + |R(x + d, y)| & \text{si } R(x - d, y) > 0 \text{ \& } R(x + d, y) < 0 \\ 0 & \text{otros casos} \end{cases} \quad (2.1)$$

Siendo R la salida del filtrado DoG.

2.2.1.2 Detector de bordes de Canny

Canny es un algoritmo [11] de múltiples etapas para detectar bordes en imágenes. Fue desarrollado por John F. Canny en 1986. Sigue 3 criterios:

- Un criterio de detección: Debe haber una baja probabilidad de fallar el marcado de píxeles reales del borde al igual que una baja probabilidad de marcar píxeles de falsos bordes.
- El criterio de localización: Establece que la distancia de los píxeles marcados como bordes debe ser mínima al centro del borde real.
- Solo debe haber una respuesta para un único borde.

El algoritmo de Canny sigue 4 fases:

- Filtrado Gaussiano: Este paso realiza la función de suavizar la imagen original para suprimir ruidos, teniendo en cuenta que un suavizado excesivo puede eliminar los bordes que se quieren detectar y un suavizado escaso puede no eliminar el ruido. Un ejemplo de una máscara gaussiana es:

$$G = \frac{1}{115} \begin{bmatrix} 2 & 4 & 5 & 4 & 2 \\ 4 & 9 & 12 & 9 & 4 \\ 5 & 12 & 15 & 12 & 5 \\ 4 & 9 & 12 & 9 & 4 \\ 2 & 4 & 5 & 4 & 2 \end{bmatrix}$$

- Obtención del gradiente: Se debe realizar para cada píxel el cálculo del módulo y orientación del gradiente mediante estas ecuaciones:

$$|G| = \sqrt{Gx^2 + Gy^2}$$

$$\theta = \text{atan}\left(\frac{Gy}{Gx}\right)$$

Siendo Gx y Gy matrices que indican la primera derivada en horizontal, Gx, y en vertical, Gy que se pueden obtener por ejemplo usando máscaras de Sobel o Prewitt.

El ángulo se redondea a la vertical, horizontal y a sus diagonales formando ángulos de 0°, 45°, 90° y 135° usando solo estas 4 direcciones.

- Selección de máximos: Con este paso se quiere conseguir un ancho de borde de un píxel suprimiendo todos los valores de gradiente excepto los máximo locales. Lo primero es comparar el píxel actual con los píxeles conectados en diferentes orientaciones; si el píxel tiene un mayor valor del módulo de gradiente con la misma dirección este valor será preservado y en caso contrario será descartado.
- Dos umbrales: Aún en el proceso puede haber bordes en la imagen causados por la variación de la iluminación, para suprimir estas respuestas se establecen 2 umbrales, alto y bajo, clasificando los píxeles de cada borde en fuertes y débiles. Si el módulo del gradiente del píxel es mayor que el umbral alto es etiquetado como píxel fuerte y si el módulo está entre el umbral alto y bajo se etiqueta como píxel débil si por el contrario su módulo es menor que el umbral bajo este píxel es suprimido. El objetivo de todo este etiquetado es rescatar los bordes débiles unidos a los fuertes.

2.2.2 Segunda derivada

En derivadas segundas se hace uso del operador laplaciano, primero definimos una aproximación discreta del operador derivada:

$$\frac{\partial f(x)}{\partial x} \approx f(x-1) + f(x)$$

En segunda derivada:

$$\frac{\partial^2 f(x)}{\partial x^2} \approx f(x+1) + f(x-1) - 2f(x)$$

Si aplicamos sobre 2 variables (x, y) obtenemos el operador laplaciano que es la suma de las derivadas parciales, aplicando el mismo proceso sobre la variable y y sumando:

$$\nabla^2 f(x, y) = \frac{\partial^2 f(x, y)}{\partial x^2} + \frac{\partial^2 f(x, y)}{\partial y^2} = f(x+1, y) + f(x-1, y) + f(x, y+1) + f(x, y-1) - 4f(x, y)$$

Gracias a esto se puede construir una máscara que aproxime el operador laplaciano:

$$W = \begin{bmatrix} 0 & 1 & 0 \\ 1 & 4 & 1 \\ 0 & 1 & 0 \end{bmatrix}$$

Ahora solo hace falta filtrar la imagen con la máscara y localizar los cruces por cero para obtener la frontera.

2.2.2.1 Laplacian of Gaussian (LoG)

Laplacian of Gaussian (LoG) [17] se define como:

$$\Delta G(x, y, \sigma) = \nabla^2 G(x, y, \sigma) = \frac{1}{\pi\sigma^4} \left(\frac{x^2 + y^2}{2\sigma^2} - 1 \right) e^{-\frac{x^2 + y^2}{2\sigma^2}}$$

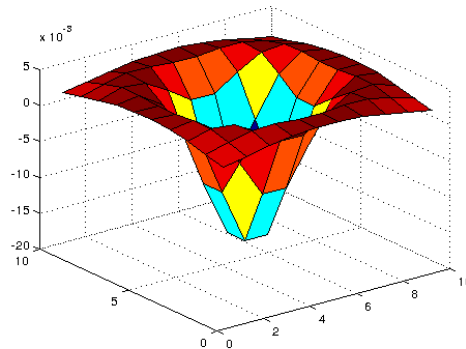


Figura 2.6: Filtro LoG con sigma 2 y tamaño 10x10 generado con matlab

Este filtro es simétrico y circular y gracias a estas características puede detectar fronteras en todas las direcciones.

Al igual que en el procedimiento anterior, se debe filtrar la imagen y localizar cruces por cero para detectar las fronteras. Se expone a continuación un ejemplo con imágenes del proceso:

Primero filtrar la imagen de entrada en escala de grises:



Figura 2.7: (a) Imagen en escala grises, (b) Resultado de aplicar filtrado

Después identificamos los cruces por cero y etiquetamos como unos:



Figura 2.8: Detección de bordes usando LoG

Se debe de añadir que estos filtros se usan además a la hora de identificar objetos en diferentes escalas, a partir de la normalización del filtro o lo que es lo mismo hacer un filtro invariante. Para ello hay que añadir:

$$\text{Filtro invariante: } \sigma^2 \triangle G(x, y, \sigma)$$

2.3 Morfología matemática

La morfología matemática es una técnica de análisis y tratamiento de estructuras geométricas para el estudio de la forma, la conectividad y el tamaño. Originalmente la morfología matemática fue desarrollada para imágenes binarias y después se extendió a imágenes en escala de grises.

La información obtenida como resultado de estas técnicas, depende del elemento estructurante que se use, ya que este puede tener diferente tamaño, forma y conectividad. Hay que caracterizarlos en función de la necesidad. Lo importante en los elementos estructurantes es definir su centro, píxel central, ya que este será la posición donde se establezca la operación.

Siguen algunos ejemplos:

- Línea: No es más que un vector de unos que se puede cambiar el tamaño y orientación en función de la necesidad; el siguiente ejemplo es para una línea de tamaño 5 con orientación horizontal:

$$\begin{bmatrix} 1 & 1 & \boxed{1} & 1 & 1 \end{bmatrix}$$

- Disco: Es una matriz de ceros y unos que simula un rombo y con parametro R de radio, marca la distancia en píxeles del centro a los

lados:

$$\begin{bmatrix} 0 & 0 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 0 \\ 1 & 1 & \boxed{1} & 1 & 1 \\ 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$$

2.3.1 Operaciones morfológicas binarias

Desde una perspectiva geométrica [12], la idea más importante es evaluar el elemento estructurante sobre la imagen y marcar la localización donde este elemento entra parcial o totalmente en la imagen, gracias a esto se puede obtener información estructural de los objetos de la imagen.

Se consideramos 2 tipos de imágenes binarias, euclidiana y discreta. Una imagen binaria euclidiana es un subconjunto del espacio Euclídeo de n dimensiones. Para el procesamiento de señales se considera $n = 1$ (una dimensión) y para imágenes $n = 2$ (dos dimensiones), una imagen euclidiana binaria se puede definir como un subconjunto del plano Euclídeo. Para una implementación digital una imagen binaria discreta se considera un subconjunto del plano cartesiano (x, y) .

2.3.1.1 Erosión

Lo primero es definir la traslación de un elemento A sobre el punto x en el plano Euclídeo:

$$A + x = \{a + x : a \in A\} \quad (2.2)$$

Geoméricamente, es trasladar el conjunto A de acuerdo con x .

La operación morfológica erosión [12] de A mediante el elemento estructurante B se define como:

$$A \ominus B = \{x : B + x \subset A\}$$

El siguiente ejemplo muestra teniendo 2 puntos x y z como uno cumple la operación, x , y el otro no, z .

$$B + x \subset A, x \in A \ominus B \quad y \quad B + z \not\subset A, z \notin A \ominus B \quad (2.3)$$

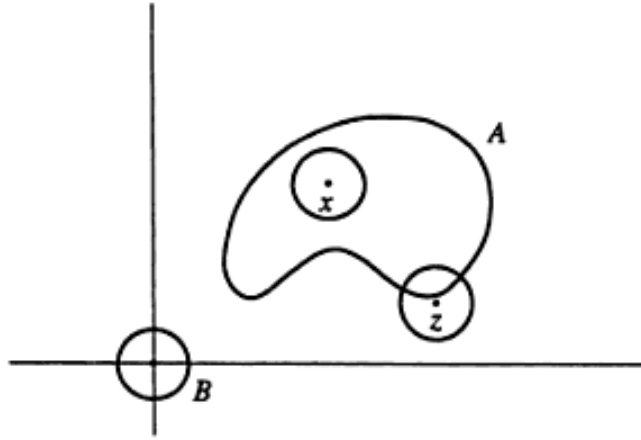


Figura 2.9: Ejemplo de operación de erosión

Geoméricamente, el disco B se debe ir moviendo alrededor y dentro de A y marcando los puntos donde B quepa.

A continuación se muestra un ejemplo para una imagen binaria desde un punto de vista digital, para ello se va a utilizar una imagen donde se puede ver las letras del alfabeto, la idea es aplicar erosión con un elemento estructurante de tipo disco para encoger las letras.

El elemento estructurante será:

$$SE_{disk} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Y se aplica la operación de erosión:



Figura 2.10: Operación de erosión binaria (a) Imagen original, (b) Resultado de aplicar erosión

Se aprecia en la figura 2.10 que las letras han encogido, si se eligiera un elemento estructurante de mayor tamaño tenderían a desaparecer las letras en la imagen.

2.3.1.2 Dilatación

La dilatación de A con un elemento estructural B está definido como:

$$A \oplus B = \bigcup A + b : b \in B \quad (2.4)$$

El proceso calcula la unión del elemento estructural B para todos los puntos de la imagen A.

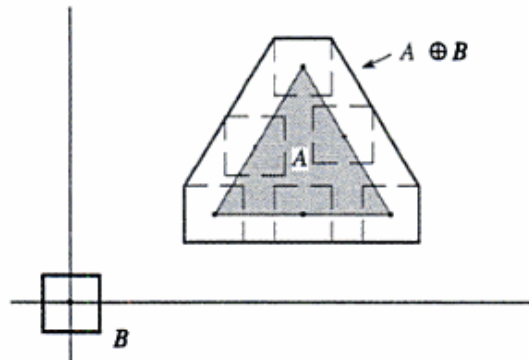


Figura 2.11: Ejemplo de operación de dilatación

A continuación se muestra un ejemplo parecido al de erosión, ya que se aplicará sobre la misma imagen con igual elemento estructurante, pero se observará que las letras se ensanchan:

El elemento estructurante será:

$$SE_{disco} = \begin{bmatrix} 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 & 0 \\ 1 & 1 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 1 & 0 \\ 0 & 0 & 1 & 1 & 0 & 0 \end{bmatrix}$$

Y se aplica la operación de dilatación:



(a)



(b)

Figura 2.12: Operación de dilatación binaria (a) Imagen original, (b) Resultado de aplicar dilatación

La finalidad de este proceso es difuminar objetos, extender los bordes, unir e incluso rellenar huecos.

2.3.1.3 Apertura y cierre

El operador de apertura se define como :

$$A \circ B = (A \ominus B) \oplus B \quad (2.5)$$

Es una combinación de erosión y dilatación ya que estos operadores son los primarios y los de apertura y cierre son secundarios, son una combinación de estos:

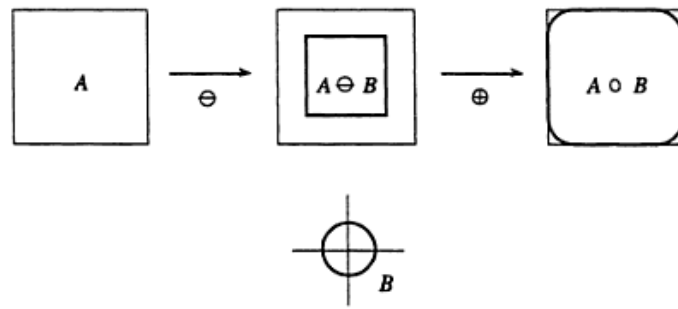


Figura 2.13: Apertura

El operador de cierre se define:

$$A \bullet B = (A \oplus B) \ominus B$$

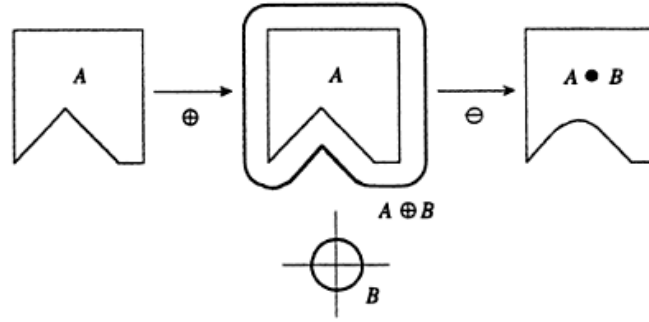


Figura 2.14: Cierre

Tienen diferentes aplicaciones, apertura para poder moldear el objeto en función del elemento estructurante y cierre para rellenar huecos sin extender excesivamente los bordes.

2.3.2 Operaciones morfológicas en escala de grises

Las operaciones morfológicas también se puede extrapolar en imágenes en escala de grises. Aquí el resultado depende de los valores de la imagen pero de diferente forma de en binaria. En binaria se buscaba que el elemento estructurante cupiera en la imagen, mientras que en escala de grises se realiza un cálculo para cada píxel, en función de la imagen de entrada y del elemento estructurante.

Se debe añadir que en escala de grises hay 2 [31] tipos de elementos estructurante, plano o no plano. Los elementos estructurante planos son los usados en morfología binaria, el nombre de plano viene dado por el perfil de intensidad que tienen. En binario el único cambio de intensidad es pasar de blanco a negro o viceversa, mientras que en escala de grises hay cambios

en el nivel de gris, por este hecho a los elementos estructurales binarios se llaman planos.

Los elementos estructurales usados para ejemplificar los operadores morfológicos en escala de grises serán planos.

2.3.2.1 Erosión

La erosión de A mediante el elemento estructurante plano B en cualquier píxel (x, y) es definido como el valor mínimo de la imagen en la región coincidente de B con A. Viene dado por:

$$A \ominus B_p(x, y) = \min_{(s, t) \in B_p} \{A(x + s, y + t)\}$$

Si escogemos un elemento estructurante no plano la erosión en cualquier píxel (x, y) es definido como:

$$A \ominus B_{np}(x, y) = \min_{(s, t) \in B_{np}} \{A(x + s, y + t) - B_{np}(s, t)\}$$

A diferencia de la operación con un elemento plano, en no plano se debe restar el valor del elemento estructurante, ya que tiene nivel de gris.

2.3.2.2 Dilatación

La dilatación de A mediante el elemento estructurante plano B, en cualquier píxel (x, y) es definido como el valor máximo de la imagen en la región coincidente de B con A. Viene dado por:

$$A \oplus B_p(x, y) = \max_{(s, t) \in B_p} \{A(x - s, y - t)\}$$

Para entender mejor la ecuación de arriba se debe definir la reflexión de un elemento B denotado como \hat{B} y definido:

$$\hat{B}(x, y) = B(-x, -y)$$

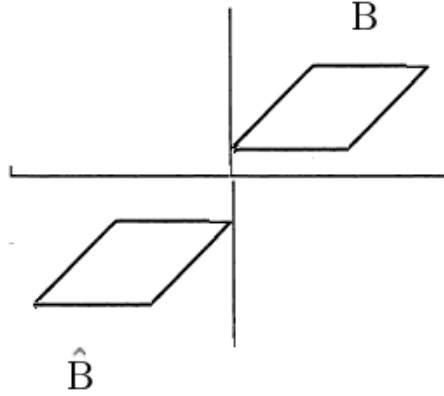


Figura 2.15: Ejemplo de reflexión

La dilatación se puede ver igual que la erosión, pero eligiendo el valor máximo y sabiendo que el elemento estructurante está reflejado.

Si aplicamos con elemento estructurante no plano, la dilatación se define como:

$$A \oplus B_{np}(x, y) = \max_{(s,t) \in B_{np}} \{A(x-s, y-t) + B_{np}(s, t)\}$$

2.3.2.3 Apertura y cierre

La operación de apertura de la imagen A mediante el elemento B se define como:

$$A \circ B = (A \ominus B) \oplus B \quad (2.6)$$

Y la operación de cierre:

$$A \bullet B = (A \oplus B) \ominus B$$

Teniendo la imagen de ejemplo:



Figura 2.16: Ejemplo para los operadores de apertura y cierre en escala de grises, Lenna

Aplicamos apertura y cierre para la imagen:

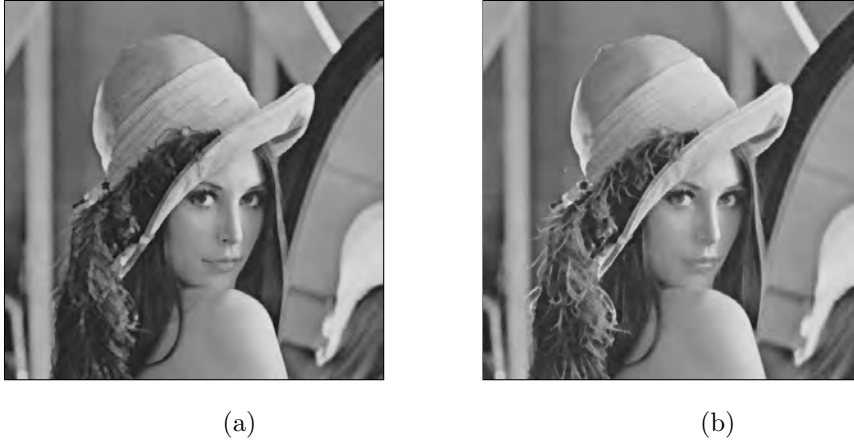


Figura 2.17: (a) Aplicando apertura mediante elemento estructurante tipo disco de radio 2, (b) Aplicando cierre mediante elemento estructurante tipo disco de radio 2

Estos operadores, si se usa un elemento estructurante plano, tienden a suavizar el nivel de gris de la imagen. Se ve que en apertura baja el valor de los niveles de gris máximos y en cierre aumenta el valor de los niveles de gris mínimos. Todo esto depende del elemento estructurante.

2.3.2.4 Transformación “top-hat” y “bottom-hat”

Estas transformaciones usan la resta de la imagen con el resultado de la operación de apertura o cierre aplicadas a la imagen.

La transformación “top-hat” de la imagen A en escala de grises, mediante el elemento estructurante B se define como:

$$T_{hat}(A) = A - (A \circ B)$$

La transformación “bottom-hat” de la imagen A en escala de grises mediante el elemento estructurante B se define como:

$$B_{hat}(A) = (A \bullet B) - A$$

Es la resta del resultado de aplicar cierre en la imagen A con A.

La principal aplicación de estas transformaciones es devolver objetos de la imagen mediante el uso de elementos estructurales que han sido borrado al aplicar la operación de apertura o cierre.

Si tenemos la imagen:

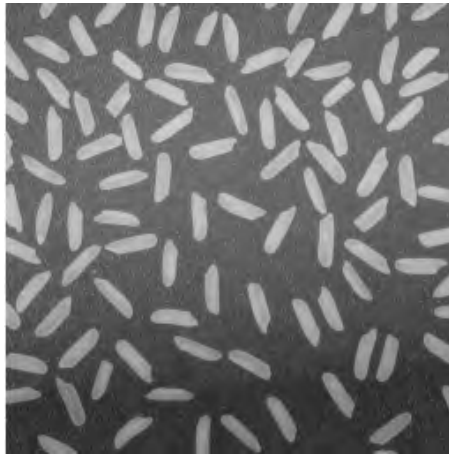


Figura 2.18: Imagen con granos de arroz

Los granos de arroz en la imagen tienen un ancho medio de 8 píxeles, si se establece como elemento estructurante B de tipo cuadrado y lado 8, los granos de arroz de mas de 8 píxeles de ancho permanecerán en la imagen y los de menos serán suavizados. En la transformación “top-hat” pasará lo contrario:

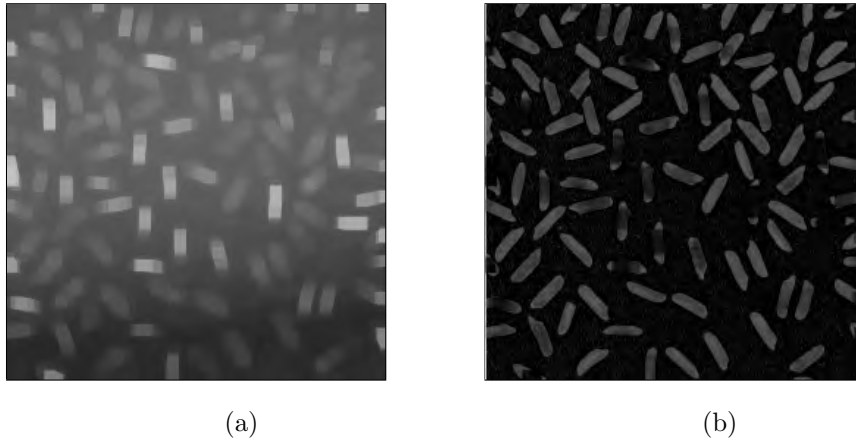


Figura 2.19: (a) Aplicando apertura mediante elemento estructurante tipo cuadrado de lado 8, (b) Aplicando “top-hat” mediante elemento estructurante cuadrado de lado 8

Los objetos que mas resaltan en apertura resaltan menos en la transformación, se demuestra lo explicado anteriormente.

Si se quiere aplicar la transformación de “bottom-hat” la imagen debe ser la contraria a la que se usó como ejemplo, los objetos sean oscuros y el fondo claro. Para al aplicar la operación de cierre se difuminen los objetos oscuros de menor ancho a 8 píxeles.

Y comparamos:

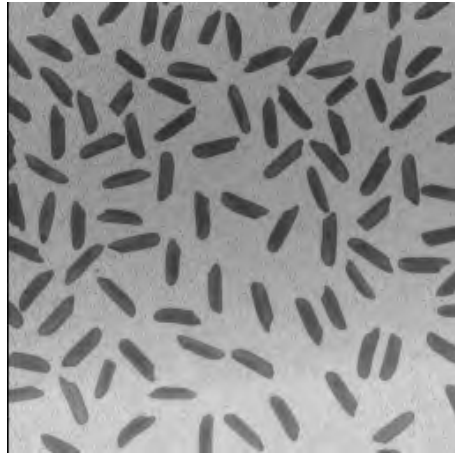
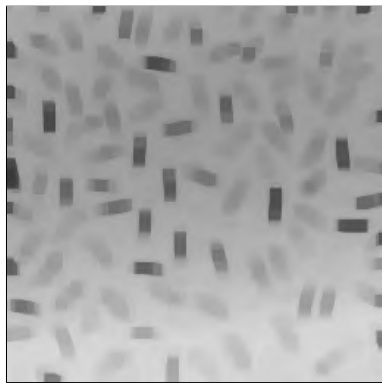
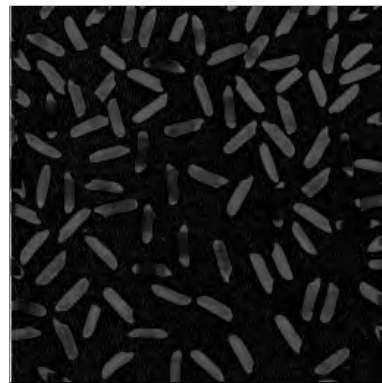


Figura 2.20: Imagen con granos de arroz



(a)



(b)

Figura 2.21: (a) Aplicando cierre mediante elemento estructurante tipo cuadrado de lado 8, (b) Aplicando “bottom-hat” mediante elemento estructurante uadrado de lado 8

2.4 Apertura de área

Este operador elimina de una imagen binaria objetos conectados con un área menor de un parametro λ [13]

$$\gamma_{\lambda}^a = \{x \in X | Area(C_x(X)) \leq \lambda\}$$

A continuación se muestra un pequeño ejemplo, la siguiente figura es de una montaña de sal con pequeños granos alrededor, este método borra elementos pequeños de la imagen, lo pequeños granos de fuera.

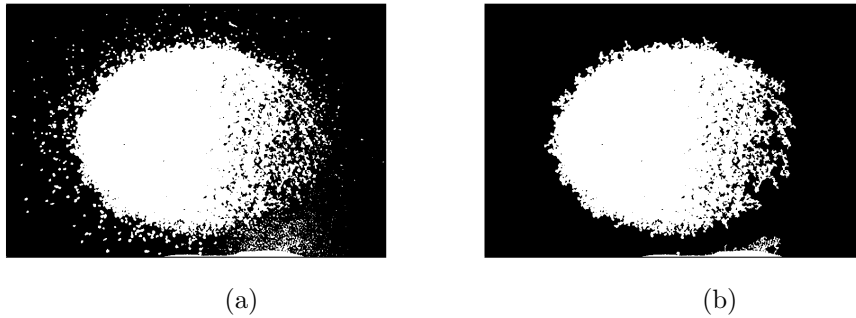


Figura 2.22: (a) Imagen original, (b) Apertura de área binaria

Este algoritmo [14] si se aplica a una imagen binaria, sigue estos pasos:

- Determinar los componentes conectados en función de la conectividad deseada.
- Computar el área de cada componente.
- Borrar los componentes con menor área elegida

Capítulo 3

Estado del arte

En este apartado se van a mencionar diferentes propuestas que abarcan la solución al problema o por lo menos tienen relación con parte de los objetivos del proyecto. Los expuestos en este capítulo son los que se han considerado más relevantes, pero se puede encontrar más información en los siguientes documentos: [18], [25], [26] y [27]

3.1 Dull Razor

Es un software [3] de segmentación de pelo y borrado del mismo. La idea principal de este software, es hacer un borrado de los pelos que puede contener la imagen sin llegar a modificar cualquier otro objeto, como un lunar. Este software se puede usar en estudios de diagnóstico de melanoma para un preprocesado que elimina los pelos y quedarse con lo esencial.

El software DullRazor sigue estos pasos:

1. Identificar la localización de los pelos oscuros: El software usa el espacio de colores RGB para leer las imágenes, después se aplica la operación morfológica de cierre en los 3 canales por separado(R,G,B). Esta operación suaviza los valores bajos de intensidad, es decir, los píxeles de pelos oscuros a lo largo de la dirección del elemento estructurante.

Los resultados experimentales sugieren que con 3 elementos estructurales en diferentes direcciones, 0° , 45° y 90° es suficiente para suavizar todos los pelos oscuros. Los elementos estructurantes están expuestos en la siguiente figura:

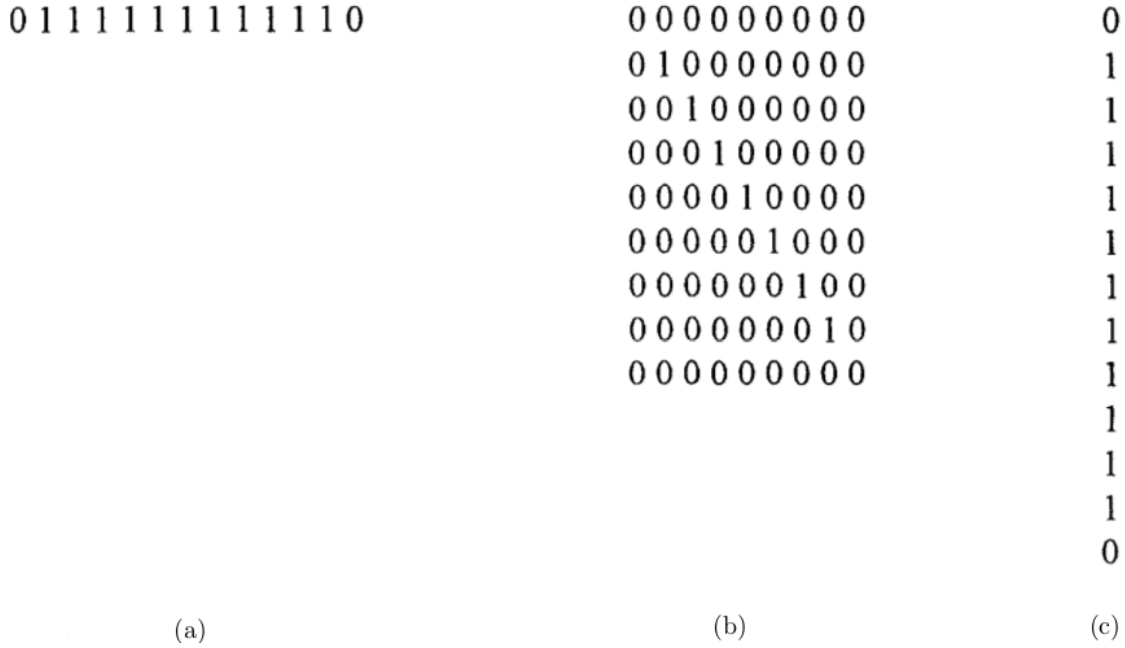


Figura 3.1: Elementos estructurantes para la operación de cierre en escala de grises (a) Elemento estructurante horizontal centrado en (6,0), (b) Elemento estructurante en diagonal centrado en (4,4), (c) Elemento estructurante vertical centrado en (0,6)

La imagen general de cierre se obtiene, cogiendo los valores máximos para los 3 resultados de aplicar la operación en las diferentes direcciones. Finalmente la máscara de pelo se obtiene estableciendo un umbral sobre la diferencia entre la imagen de la componente de color con la imagen general de cierre en valor absoluto.

La imagen general de cierre para una componente se define como, ejemplo para la componente R:

$$G_r = |O_r - \max\{O_r \bullet S_0, O_r \bullet S_{45}, O_r \bullet S_{90}\}| \quad (3.1)$$

Siendo O_r la imagen original en componente R, S_0 el elemento estructurante horizontal, S_{45} el elemento estructurante diagonal, S_{90} en

vertical y \bullet significa la operación de cierre.

La máscara binaria de pelo se identifica como:

$$M_r(x, y) = \begin{cases} 1 & \text{si } G_r(x, y) < T \\ 0 & \text{resto casos} \end{cases} \quad (3.2)$$

Donde T es un valor de umbral predeterminado.

Comor resultado final la máscara binaria general en función de las 3 componentes se expresa como la unión de las 3.

$$M = M_r \cup M_g \cup M_b \quad (3.3)$$

Donde M_r , M_g y M_b son las máscaras binarias correspondientes a las componentes R, G y B.

2. Interpolación: La máscara binaria obtenida en el paso anterior. es usada como guía para reemplazar los píxeles de la imagen original, donde se ha detectado un pelo por valores de píxeles cercanos detectados como no pelo.

Para cada píxel dentro de la región de pelo de la máscara M, se pintan líneas en ocho direcciones, arriba, abajo, derecha, izquierda y las cuatro diagonales extendiéndose desde el píxel hasta que la línea alcanza región de no pelo.

Se calcula la logitud de estas líneas y la mas larga es anotada. La línea mas larga debe ser mayor que 50 píxeles y las otras líneas mas cortas de 10 píxeles, si no es así este píxel es rechazado.

Cuando este píxel es verificado de estar en una estructura de pelo, se reemplaza usando interpolación binaria, por el valor de 2 píxeles cercanos a lo largo de la línea corta, línea perpendicular a la larga.

Si denominamos el valor de la intensidad de un píxel a reemplazar como $I(x,y)$ y los píxeles usados para la interpolación como $I_1(x_1,y_1)$ y $I_2(x_2,y_2)$. El nuevo valor de intensidad se obtendra como $I_n(x,y)$:

$$I_n(x,y) = I_2(x_2,y_2) \frac{D(I(x,y), I_1(x_1,y_1))}{D(I_1(x_1,y_1), I_2(x_2,y_2))} + I_1(x_1,y_1) \frac{D(I(x,y), I_2(x_2,y_2))}{D(I_1(x_1,y_1), I_2(x_2,y_2))}$$

Donde

$$D(A(a,b), B(c,d)) = \sqrt[2]{(c-a)^2 + (d-a)^2}$$

3. Normalmente la localización de los bordes no suele estar bien trazada, por el efecto del ruido y el efecto de penumbra de los pelos y se forman líneas cerca de los bordes de las regiones de los pelos. Para descartar este efecto se aplica el operador de dilatación con un elemento estructurante tipo cuadrado con todos los elementos a uno y a un filtro adaptativo de mediana de 5x5 para suavizar, obteniendo una nueva máscara binaria.

A continuación se muestra un pequeño ejemplo usando DullRazor en imágenes del proyecto.

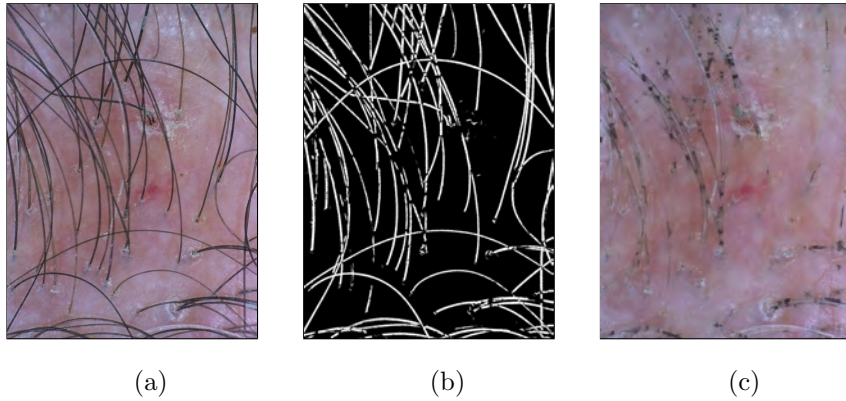


Figura 3.2: (a) Imagen original de ejemplo, (b) Mascara aplicando DullRazor (c) Aplicando interpolación

Se debe de añadir que la parte de segmentación del cabello, es una buena aproximación como segmentación y por esto se usará como referencia a la hora de evaluar los diferentes métodos de segmentación mencionados durante la memoria.

3.2 TrichoScan

Es un software [4] que estudia el cabello haciendo uso parámetros biológicos como la densidad, el diámetro y la tasa de crecimiento. Para ello se llevan a cabo estos precedimientos [5]:

Durante la primera sesión, se elige la zona que se va a explorar —una superficie de 2 centímetros de diámetro, que presente el problema, pero que no sea visible con el peinado habitual del paciente. Se rasura y se tiñe el pelo, para asegurarse de captar y contar también el pelo más fino o más claro.

En la segunda sesión se toma una fotografía a 20 aumentos de la zona y se analizan los resultados. Esta sesión se realiza tres días después de la primera para distinguir el cabello que ha crecido —en fase de anágeno— del

cabello que no — telógeno.



Figura 3.3: Ejemplo de imagen de TrichoScan

3.3 FotoFinder Trichoscale

FotoFinder Trichoscale [29] es un nuevo estandar para cuantificar la crecida y la pérdida de cabello.

La evaluación se realiza en un aera pequeña de 5mm tiñiendo la zona, después la foto es capturada por una ca cámara médica y evaluada por el software.

El software realiza la medición de estos parámetros:

- Cantidad y densidad de pelo.
- Esquema de la proporción de cabellos anágenos y telógenos.
- Medición de la unidades foliculares.

3.4 Segmentación de los vasos sanguíneos de la retina

La segmentación de los vasos sanguíneos se usa para el diagnóstico de la retinopatía diabética, provoca alteraciones en los vasos sanguíneos de la retina y puede derivar en ceguera si no se diagnostica a tiempo.

El estudio de la segmentación de los vasos sanguíneos de la retina está relacionada con la segmentación de pelo por la forma que tienen; estructuras alargadas y estrechas con intersecciones entre ellas.

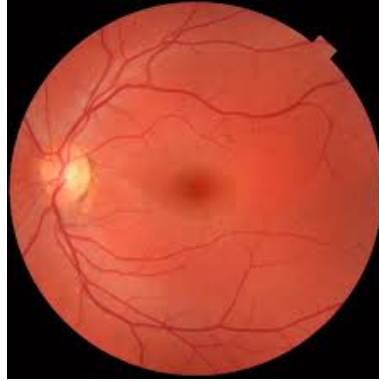


Figura 3.4: Ejemplo de vasos sanguíneos de la retina [23]

En el siguiente documento [24] se citan varios métodos usados para segmentar vasos sanguíneos haciendo una comparativa final de todos. Podemos destacar:

- “Match Filtering”: Se basan en unas observaciones previas, los vasos sanguíneos tienen curvatura limitada, el diámetro de éstos decrece en la medida en que te alejas del disco óptico y la intensidad de los píxeles en la sección transversal de los vasos sanguíneos siguen un perfil gaussiano.

Para la implementación del algoritmo de match 2D se han usado 12 filtros para cubrir todas las direcciones con una σ sobre la componente G de RGB:

$$N = \{(u, v) | |u| \leq 3\sigma, |v| \leq \frac{L}{2}\} \text{ donde } L = 9$$

Estos filtros gaussianos son utilizados como máscaras de convolución y el filtro que dé la máxima respuesta será considerado la orientación

del vaso.

- Reconocimiento de patrones: Usan la componente G de RGB y para el preprocesado de la imágenes usan la operación de apertura con elemento estructurante de tipo disco con diametro igual a tres para las zonas más claras de la parte central de los vasos, un filtro de media de 69x69 para crear un fondo homogeneo de nivel de gris y por último la transformación “top-hat” usando 8 píxeles de radio con elemento estructurante tipo disco para mejorar las zonas oscuras, incluyendo los vasos sanguíneos.

Como paso principal, después del preprocesado, usan una red neuronal para clasificar los píxeles como vaso o no. Para ello usan un vector de características basado en la información a nivel local del nivel de gris.

- Detector de líneas multiescala: Para el preprocesado sigue el mismo método expuesto en el anterior procedimiento. Para detectar las vasos como líneas usan diferentes longitudes, 3,5,...,15. Para cada longitud y cada píxel se calcula la media del nivel de gris con ventana 15x15 que se llamará $I^{avg}(x,y)$. Después para cada longitud se computa una media ponderada de los valores de gris a largo de la línea para 18 direcciones, 0, 10..., 170 grados, la mayor respuesta, I^s , de todas las direcciones se calcula para cada píxel. La respuesta para cada escala es calculada como $R^s = I^s - I^{avg}$ y se reescalada para tener media cero y varianza unidad, \hat{R}^s .

Para obtener la respuesta multiescala se computa una combinación lineal de las respuestas:

$$R = \frac{1}{64}(\sum ss\hat{R}^s + I)$$

Como paso final se usa el método de Otsu para binarizar la imagen.

- “Scale-space analysis”: Lo que se pretende es evaluar el gradiente y el hessiano para cada píxel y escala, para obtener una clasificación que combine todas las escalas. El gradiente y el hesseano se obtiene convolucionando la imagen de entrada $I(x,y)$ con las máscaras correspondientes:

$$I_x = I(x, y) * sG_x$$

$$I_y = I(x, y) * sG_y$$

$$I_{xx} = I(x, y) * s^2G_{xx}$$

$$I_{xy} = I(x, y) * s^2G_{xy}$$

$$I_{yy} = I(x, y) * s^2G_{yy}$$

G es el kernel guassiano con desviación típica s y el subíndice indica derivada parcial.

Los bordes de los vasos vendrán caracterizados por la respuesta del gradiente y el centro vendrá por el hessiano. Las escalas de s van desde $s_{min} = 1,5$ hasta $s_{max} = 10$ incrementando de 0,5 en 0,5.

- Procesado morfológico: Para el elemento estructurante se elige un elemento circular por el hecho de que los vasos no tienen dirección preferente.

Como la operación de top-hat es muy sensible al ruido se usa un operador modificado:

$$tophat(img) = img - min(open(close(img, Wc), W), img)$$

El elemento Wc es el asociado a la operación de cierre, el cual debe ser más estrecho y más largo que el típico ruido que se desea eliminar. W

es elegido como un disco variando su radio de 1 píxel hasta 8 píxeles; esto da ocho resultados.

Para el resultado final, se obtiene la respuesta mayor de los ocho resultados para cada píxel, al final se normaliza la salida obteniendo valores entre 0 y 1.

Capítulo 4

Diseño técnico

En este capítulo se exponen los diferentes procedimientos para obtener la mejor solución al problema. En la siguiente figura se expone un diagrama de flujo del funcionamiento del sistema:

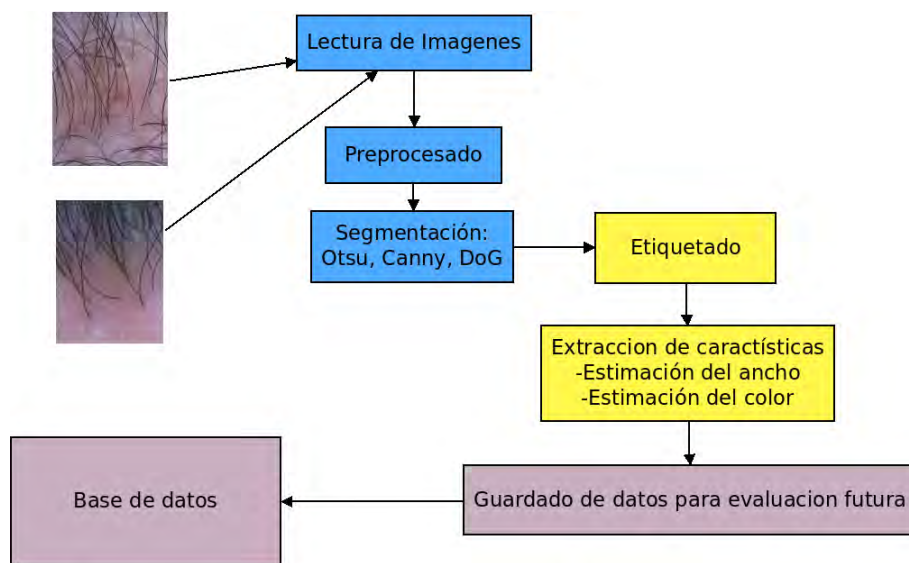


Figura 4.1: Flujograma del sistema

4.1 Preprocesado

Para mejorar la imagen de entrada y poder trabajar con ella, es importante realizar unos pasos previos que mejoren la calidad de la imagen y facilitar la segmentación posterior.

El preprocesado se divide en 2 partes, una transformación de la imagen a escala de grises para ser usada durante todo el proyecto y de un filtrado para eliminar ruido. Este paso es muy importante ya que los resultados de la segmentación dependen de un buen preprocesado.

4.1.1 Componentes de colores

Es importante elegir bien una componente de color para trabajar ya que cada espacio de color tiene diferentes dependencias y aún más sus componentes o dimensiones.

En este proyecto se va a trabajar con 2 componentes:

- Componente L de CIELAB: Su uso [18] adapta el algoritmo a la percepción humana, porque la luminancia coincide con la respuesta perceptual de la luz del sistema visual humano.



Figura 4.2: Imagen del proyecto en componente CL

- Componente principal: Maximiza la varianza en función de las 3 componentes de RGB:

$$C = w_R R + w_G G + w_B B$$

Donde los factores w forman parte de un vector unitario \hat{W} es obtenido del autovector dominante (asociado al mayor autovalor) de la matriz de covarianzas de las 3 componentes.

Los autovalores y autovectores se obtienen diagonalizando la matriz de covarianzas de este modo:

$$M_d = \begin{bmatrix} d1 & 0 & 0 \\ 0 & d2 & 0 \\ 0 & 0 & d3 \end{bmatrix} = P^{-1} M_{cov} P$$

Donde los autovectores serán las columnas de P , y los autovalores serán los valores de la diagonal principal de M_d . \hat{W} será igual al autovector asociado al máximo autovalor.



Figura 4.3: Imagen del proyecto en componente CP

$$Var(CP) = 2236,5$$

- "Zero Padding": Se rellena con ceros los bordes de la imagen hasta que la ventana puede centrarse en la primera posición de la imagen(1,1). Si los bordes son importantes este metodo no es conveniente, ya que en los bordes estas añadiendo información independiente de la imagen.
- Envolver la imagen: Considerar los píxeles contiguos como padding y realizar la operación. La siguiente figura muestra un pequeño ejemplo si se tiene un ventana de 3x3:

60	17	29				
31	10	12	45	70	31	
8	52	12	25	11	8	
			6	7	32	10
			17	46	50	2
			17	29	67	70
						60

Figura 4.4: Padding con información de los pixeles contiguos

Antes de explicar los tipos de filtros de ordenamiento estadísticos se va a ejemplificar el procedimiento para el píxel central de una matriz. Teniendo de entrada la siguiente matriz:

$$\begin{bmatrix} 10 & 12 & 45 & 70 & 31 \\ 52 & 12 & 25 & 11 & 8 \\ 6 & 7 & 32 & 10 & 78 \\ 17 & 46 & 50 & 2 & 27 \\ 17 & 29 & 67 & 70 & 60 \end{bmatrix}$$

Y escogiendo una ventana de 3x3 y centrandola en el píxel central obtenemos:

$$\begin{bmatrix} 10 & 12 & 45 & 70 & 31 \\ 52 & \boxed{12} & \boxed{25} & \boxed{11} & 8 \\ 6 & 7 & 32 & 10 & 78 \\ 17 & \boxed{46} & \boxed{50} & \boxed{2} & 27 \\ 17 & 29 & 67 & 70 & 60 \end{bmatrix}$$

Se ordenan los valores:

$$\begin{bmatrix} 2 & 7 & 10 \\ 11 & 12 & 25 \\ 32 & 46 & 50 \end{bmatrix}$$

Tipos de filtros estadísticos de ordenamiento:

- Filtro de mediana: Este filtro establecería a la salida, el píxel central del subconjunto ordenado de valores al establecer la ventana sobre la componente de entrada. Ejemplo

Se ordenan los valores:

$$\begin{bmatrix} 2 & 7 & 10 \\ 11 & \boxed{12} & 25 \\ 32 & 46 & 50 \end{bmatrix}$$

El el valor del nuevo píxel será el 12 ya que coincide con la mediana de este conjunto de valores, valor que está en el centro.

Este tipo de filtro se usa para eliminar ruido de tipo sal y pimienta, preserva y define mejor los bordes. En las siguientes figuras se demuestra como se puede eliminar el ruido de sal y pimienta con este tipo de filtros:



(a)



(b)

Figura 4.5: (a) Imagen con ruido, (b) Resultado de aplicar filtro de mediana

- Filtro de mínimo: Escoger el valor mínimo para el conjunto de datos

al pasar la ventana sobre la imagen. Tiende a oscurecer la imagen:

$$\begin{bmatrix} \boxed{2} & 7 & 10 \\ 11 & 12 & 25 \\ 32 & 46 & 50 \end{bmatrix}$$

- Filtro de máximo: Escoger el valor máximo para el conjunto de datos al pasar la ventana sobre la imagen. Tiende a aclarar la imagen:

$$\begin{bmatrix} 2 & 7 & 10 \\ 11 & 12 & 25 \\ 32 & 46 & \boxed{50} \end{bmatrix}$$

En este proyecto se usará un filtro de mediana y como tratamiento de los bordes se usa la opción de envolver la imagen; para realzar los bordes de los pelos y eliminar la influencia impuesta por la iluminación que puede interferir en la respuesta de la segmentación



Figura 4.6: Influencia de la iluminación

Lo fundamental aquí es escoger bien el tamaño de la ventana, una ventana pequeña puede que no suavice lo suficiente y grande suaviza tanto que puede unir intersecciones de pelos.

En el siguiente ejemplo se muestra el filtro con diferentes ventanas, con una ventana de 3x3 aún se puede apreciar la influencia de la iluminación luz, pero si se aumenta es casi inapreciable.

Subir el tamaño de la ventana también tiene sus efectos negativos; se pueden juntar las esquinas en la medida en que se va agrandando la ventana. En el siguiente ejemplo se ilustra bien lo comentado:

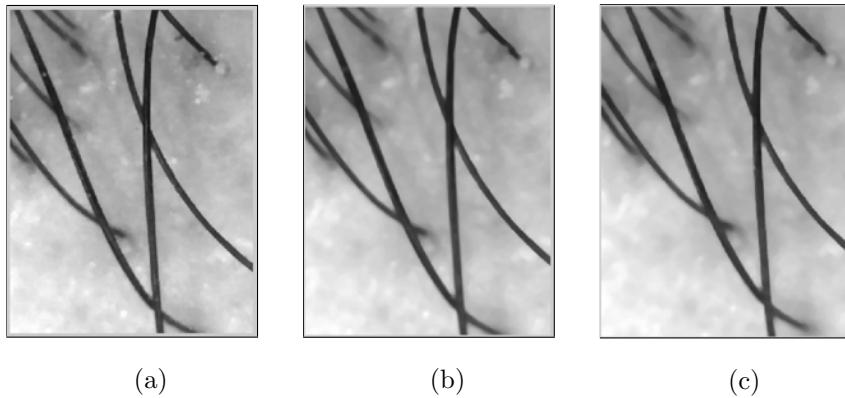


Figura 4.7: (a) Ventana 3x3, (b) Ventana 4x4 y (c) Ventana 5x5

Viendo estos resultados lo ideal es escoger una ventana de 4x4 para reducir el efecto de la iluminación y no juntar demasiado los cruces, es la opción mas equilibrada.

Otra opción podía ser usar un filtro gaussiano para difuminar el efecto de la luz, pero eso difumina tanto los bordes que es muy difícil de recuperar o no se recupera ni con un filtro de mediana, esta opción queda descartada.

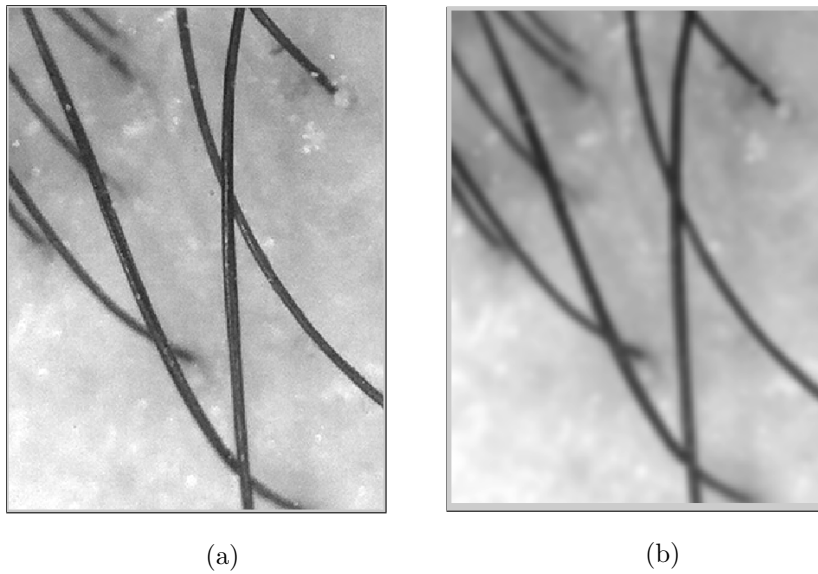


Figura 4.8: (a) Ruido por luz, (b) Filtrado gaussiano sigma 3

4.2 Segmentación

En esta sección se van a exponer los diferentes métodos de segmentación usados en el proyecto.

4.2.1 Método de Otsu

Habiendo explicado previamente el funcionamiento de Otsu, se va a aplicar este método en las imágenes dermatoscópicas del proyecto:

Como se puede apreciar en la figura 4.9, en la parte de abajo de la imagen el nivel de gris es mayor, mas claro, y en la parte superior es menor, mas oscuro. El método de Otsu no puede aplicar un umbral para hacer una buena segmentación. Una posible solución es no aplicar este método a nivel global de la imagen y aplicar a un nivel local, fragmentar la imagen en bloques cuadrados más pequeños y con estos aplicar este método.

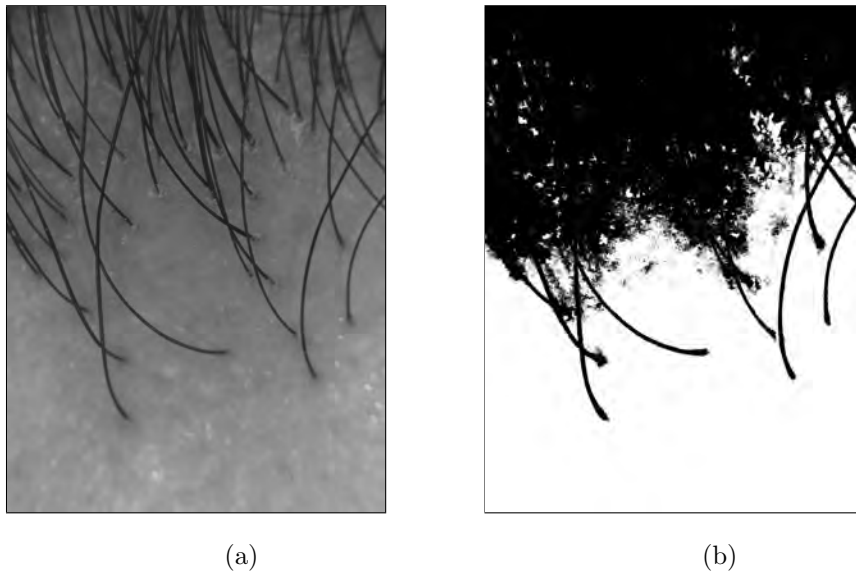


Figura 4.9: (a) Imagen en escala grises, (b) Resultado de aplicar el método de Otsu

Lo importante es elegir un paso, número de píxeles, para fragmentar la imagen teniendo en cuenta el siguiente criterio: se debe escoger un paso lo suficientemente pequeño como para el nivel de gris del vello Otsu lo pueda asociar a una clase y lo suficientemente grande para que puedan caber 2 clases, pelo y no pelo. Para la aplicación se escogió un paso de 100, fragmentando la imagen en pequeños bloques de 100x100 píxeles.

Aplicando lo anterior estos son los resultados:

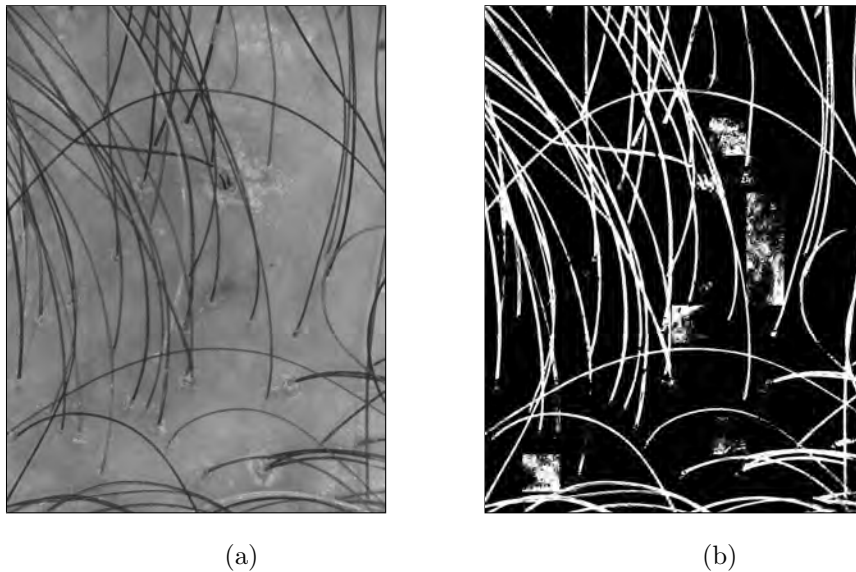


Figura 4.10: (a) Imagen en escala grises, (b) Otsu a nivel local

La fragmentación no es perfecta y genera zonas sin influencia de vello y se aprecia solo una clase ya que no hay influencia de un nivel de gris bajo, obteniendo una respuesta no deseada la cual hay que saber descartar, para mejorar la segmentación. Como solución se plantea usar el factor de calidad o usar un umbral lo suficientemente alto para asegurar que en ese trozo no hay influencia de pelo.

Se tratará el umbral de Otsu y su factor de calidad como matrices, ya que al segmentar la imagen cada segmento será un elemento de la matriz.

4.2.1.1 Factor de calidad

Como bien se definió anteriormente, el factor de calidad indica lo acertado que puede ser la segmentación y para ver su funcionalidad se aplicó en las imágenes considerando sus teniendo en cuenta sus componentes(CL,CP). La siguiente figura se muestra las matrices del factor de calidad:

Las celdas con un circulo son las celdas que representan en la imagen

0.7637	0.5634	0.6999	0.8213	0.8521	0.761	0.5679	0.6869	0.6429	0.8604	0.722
0.771	0.8526	0.7462	0.8663	0.8675	0.88	0.5448	0.6355	0.8673	0.8501	0.8777
0.8212	0.8701	0.8471	0.8706	0.8485	0.8405	0.7639	0.8317	0.8183	0.8703	0.6254
0.7465	0.8286	0.8212	0.8119	0.7578	0.8099	0.5195	0.7447	0.8535	0.8833	0.8905
0.7548	0.7444	0.7879	0.7047	0.6895	0.7071	0.6738	0.509	0.8288	0.876	0.8823
0.7611	0.7793	0.8389	0.7615	0.6559	0.7045	0.6017	0.5654	0.8532	0.8535	0.6147
0.7166	0.7992	0.7057	0.7941	0.7029	0.8269	0.7926	0.437	0.8255	0.6929	0.8366
0.7516	0.7311	0.5668	0.7761	0.7146	0.5814	0.7305	0.508	0.7513	0.8596	0.8267
0.7989	0.7387	0.7506	0.7685	0.7259	0.599	0.6934	0.8093	0.5637	0.8841	0.6435
0.8619	0.7753	0.7299	0.8074	0.733	0.677	0.6792	0.7344	0.8963	0.8347	0.8882
0.5153	0.7614	0.7035	0.7125	0.6984	0.5905	0.7231	0.5508	0.4762	0.8308	0.8484
0.7831	0.6595	0.6117	0.7445	0.6806	0.4359	0.6169	0.7564	0.8087	0.7861	0.7555
0.7651	0.568	0.6532	0.8479	0.8473	0.6353	0.5721	0.794	0.7407	0.7331	0.7821
0.8075	0.8332	0.8389	0.8425	0.837	0.8192	0.7175	0.7153	0.6566	0.6978	0.7332

Figura 4.11: Factor de calidad CL

0.7603	0.608	0.7136	0.8288	0.8522	0.7982	0.8081	0.7991	0.7797	0.8702	0.7295
0.7712	0.873	0.7329	0.8652	0.8674	0.9034	0.733	0.8943	0.9056	0.8789	0.9208
0.8259	0.8959	0.8463	0.8632	0.8522	0.8487	0.8404	0.8767	0.8985	0.8796	0.7972
0.7775	0.8315	0.8197	0.8053	0.7701	0.8128	0.5607	0.8497	0.9331	0.8953	0.8947
0.7901	0.7549	0.7835	0.7082	0.7125	0.7288	0.7699	0.6	0.9079	0.8986	0.8953
0.8019	0.7826	0.8386	0.7644	0.6919	0.7545	0.7345	0.727	0.8955	0.9114	0.6932
0.7458	0.7977	0.7125	0.7974	0.7165	0.8221	0.7949	0.45	0.8403	0.8061	0.8546
0.7652	0.7373	0.5731	0.7719	0.7368	0.5754	0.7497	0.537	0.7868	0.9178	0.9189
0.8207	0.7471	0.7474	0.7644	0.7331	0.6284	0.7111	0.8206	0.6432	0.9306	0.8333
0.8781	0.7773	0.7333	0.8145	0.7461	0.668	0.7079	0.7997	0.8992	0.8602	0.9228
0.5646	0.7631	0.6993	0.7246	0.7076	0.6518	0.7552	0.6458	0.7417	0.8776	0.8645
0.7977	0.7009	0.6427	0.7497	0.7583	0.856	0.7671	0.7856	0.8338	0.8503	0.8853
0.8047	0.585	0.6739	0.8833	0.8748	0.7593	0.7602	0.859	0.8476	0.8165	0.8369
0.8442	0.8392	0.8353	0.8393	0.873	0.8638	0.8477	0.8327	0.7802	0.7794	0.7596

Figura 4.12: Factor de calidad CP

original, un bloque donde no hay influencia de pelo, las cuales se quiere descartar del proceso.

Una forma, es coger el valor máximo de los factores de calidad ,que se sabe que no tienen influencia de pelo y eliminar los valores de las celdas que estén por debajo de este valor de umbral. Para ejemplificar este proceso se va a marcar con un rectángulo las celdas que se verían afectadas, y por lo tanto se descartarían.

Se puede ver facilmente en las figuras 4.13 y 4.14, hay varias celdas con rectángulo que se descartarían si se aplica este proceso, como consecuencia daría la perdida de información relevante para la segmentación. Por todo esto el factor de calidad se descarta como solución al problema.

0.7637	0.5634	0.6999	0.8213	0.8521	0.761	0.5679	0.6869	0.6429	0.8604	0.722
0.771	0.8526	0.7462	0.8663	0.8675	0.88	0.5448	0.6355	0.8673	0.8501	0.8777
0.8212	0.8701	0.8471	0.8706	0.8485	0.8405	0.7639	0.8317	0.8183	0.8703	0.6254
0.7465	0.8286	0.8212	0.8119	0.7578	0.8099	0.5195	0.7447	0.8535	0.8833	0.8905
0.7548	0.7444	0.7879	0.7047	0.6895	0.7071	0.6738	0.509	0.8288	0.876	0.8823
0.7611	0.7793	0.8389	0.7615	0.6559	0.7045	0.6017	0.5654	0.8532	0.8535	0.6147
0.7166	0.7992	0.7057	0.7941	0.7029	0.8269	0.7926	0.4375	0.8255	0.6929	0.8366
0.7516	0.7311	0.5668	0.7761	0.7146	0.5814	0.7305	0.5085	0.7513	0.8596	0.8267
0.7989	0.7387	0.7506	0.7685	0.7259	0.599	0.6934	0.8093	0.5637	0.8841	0.6435
0.8619	0.7753	0.7299	0.8074	0.733	0.677	0.6792	0.7344	0.8963	0.8347	0.8882
0.5153	0.7614	0.7035	0.7125	0.6984	0.5905	0.7231	0.5508	0.4762	0.8308	0.8484
0.7831	0.6593	0.6117	0.7445	0.6806	0.4359	0.6169	0.7564	0.8087	0.7861	0.7555
0.7651	0.5685	0.6532	0.8479	0.8473	0.635	0.5721	0.794	0.7407	0.7331	0.7821
0.8075	0.8332	0.8389	0.8425	0.837	0.8192	0.7175	0.7153	0.6566	0.6978	0.7332

Figura 4.13: Factor de calidad CL aplicando umbral

0.7603	0.608	0.7136	0.8288	0.8522	0.7982	0.8081	0.7991	0.7797	0.8702	0.7295
0.7712	0.873	0.7329	0.8652	0.8674	0.9034	0.733	0.8943	0.9056	0.8789	0.9208
0.8259	0.8959	0.8463	0.8632	0.8522	0.8487	0.8404	0.8767	0.8985	0.8796	0.7972
0.7775	0.8315	0.8197	0.8053	0.7701	0.8128	0.5607	0.8497	0.9331	0.8953	0.8947
0.7901	0.7549	0.7835	0.7082	0.7125	0.7288	0.7699	0.6	0.9079	0.8986	0.8953
0.8019	0.7826	0.8386	0.7644	0.6919	0.7545	0.7345	0.727	0.8955	0.9114	0.6932
0.7458	0.7977	0.7125	0.7974	0.7165	0.8221	0.7949	0.45	0.8403	0.8061	0.8546
0.7652	0.7373	0.5731	0.7719	0.7368	0.5754	0.7497	0.537	0.7868	0.9178	0.9189
0.8207	0.7471	0.7474	0.7644	0.7331	0.6284	0.7111	0.8206	0.6432	0.9306	0.8333
0.8781	0.7773	0.7333	0.8145	0.7461	0.668	0.7079	0.7997	0.8992	0.8602	0.9228
0.5646	0.7631	0.6993	0.7246	0.7076	0.6518	0.7552	0.6458	0.7417	0.8776	0.8645
0.7977	0.7009	0.6427	0.7497	0.7583	0.856	0.7671	0.7856	0.8338	0.8503	0.8853
0.8047	0.585	0.6739	0.8833	0.8748	0.7593	0.7602	0.859	0.8476	0.8165	0.8369
0.8442	0.8392	0.8353	0.8393	0.873	0.8638	0.8477	0.8327	0.7802	0.7794	0.7596

Figura 4.14: Factor de calidad CP aplicando umbral

4.2.1.2 Acotar los umbrales de gris

Primero se analizan los umbrales de gris obtenidos al aplicar el método de Otsu que se pueden ver en la siguiente figura:

0.3765	0.4157	0.3843	0.349	0.3686	0.4078	0.4941	0.3961	0.4549	0.4157	0.451
0.4039	0.4353	0.3882	0.349	0.3765	0.3765	0.545	0.498	0.4157	0.4392	0.4745
0.3961	0.4157	0.3882	0.3569	0.349	0.3725	0.4353	0.4431	0.4039	0.4392	0.5922
0.3922	0.3922	0.3686	0.3451	0.3608	0.3725	0.537	0.4627	0.4314	0.4235	0.4392
0.4039	0.4039	0.3922	0.3608	0.3608	0.3647	0.4902	0.568	0.451	0.4275	0.4078
0.3882	0.3882	0.3843	0.3569	0.3765	0.4	0.4314	0.564	0.4392	0.3961	0.4706
0.4	0.4	0.3765	0.3804	0.3961	0.3725	0.4039	0.533	0.4078	0.4824	0.4431
0.3843	0.3922	0.4157	0.3843	0.3686	0.3922	0.3804	0.525	0.4275	0.4235	0.4392
0.3725	0.3922	0.4118	0.3804	0.3765	0.478	0.3843	0.4196	0.4706	0.4196	0.6235
0.3725	0.4039	0.3882	0.3686	0.3725	0.4235	0.4431	0.4314	0.4039	0.4078	0.4275
0.4314	0.3647	0.3686	0.3765	0.4	0.451	0.4471	0.4353	0.4745	0.4353	0.4471
0.3686	0.3804	0.3882	0.3961	0.4392	0.4667	0.572	0.4	0.4078	0.4588	0.4784
0.3843	0.498	0.4235	0.4314	0.4275	0.4353	0.4431	0.3843	0.4549	0.4745	0.549
0.4	0.3843	0.3765	0.4039	0.4196	0.4	0.4196	0.4118	0.4471	0.4863	0.4902

Figura 4.15: Umbrales Otsu CL

0.6824	0.7569	0.6941	0.6275	0.6667	0.7333	0.8471	0.698	0.8039	0.7294	0.7961
0.7333	0.7843	0.702	0.6275	0.6706	0.6667	0.8588	0.851	0.7216	0.7608	0.798
0.7176	0.749	0.698	0.6471	0.6275	0.6667	0.7725	0.7725	0.7	0.7451	0.9882
0.702	0.7098	0.6667	0.6196	0.6431	0.6667	0.9294	0.8039	0.7392	0.7373	0.7686
0.7137	0.7294	0.7059	0.6431	0.6431	0.651	0.8039	0.8783	0.7804	0.7373	0.7255
0.6863	0.6941	0.6863	0.6353	0.6706	0.7137	0.7373	0.964	0.7765	0.6843	0.8392
0.7059	0.7098	0.6667	0.6745	0.698	0.6627	0.7216	0.941	0.7255	0.8471	0.7647
0.6863	0.6941	0.7333	0.6824	0.6549	0.6471	0.6784	0.937	0.7569	0.7412	0.7275
0.6667	0.702	0.7333	0.6745	0.6667	0.858	0.698	0.7451	0.8235	0.7294	0.9706
0.6706	0.7216	0.6941	0.6627	0.6627	0.7686	0.7961	0.7608	0.7216	0.7176	0.7588
0.7686	0.651	0.6627	0.6706	0.7216	0.8078	0.8039	0.7608	0.8353	0.7686	0.7804
0.6667	0.6863	0.702	0.7176	0.7765	0.8176	0.8775	0.698	0.7294	0.7922	0.8039
0.702	0.894	0.7608	0.7686	0.7608	0.7686	0.7725	0.6784	0.7922	0.8078	0.8549
0.7176	0.6902	0.6706	0.7176	0.7451	0.7137	0.7294	0.7255	0.7804	0.8118	0.851

Figura 4.16: Umbrales Otsu CP

Se sigue la misma lógica que en el punto 4.2.1.1 Factor de calidad, las celdas con un círculo indican que en el segmento original no hay influencia de pelo.

Se puede comprobar que los valores de los umbrales de gris en componente CP son mayores que CL, esto es por la característica ya comentada de CP por maximizar la varianza.

Otra forma, es aplicar una cota superior sobre los umbrales de gris, por el cual los umbrales de gris obtenidos mayores a esta cota quedan descartados. La cota tiene que ser lo suficientemente alta como para no descartar zonas con vello y lo suficientemente bajo para descartar zonas sin vello.

Analizando los niveles y comparando, se decidió usar para este método la componente CP al ser mas facil elegir una cota de descarte, esta cota tiene un valor de 0,9 que si lo llevamos a nivel de gris es un nivel de 229.5 (practicamente blanco), obteniendo este resultado:

Como se ve en la figura 4.16, aun no está completo el proceso ya que quedan umbrales con un circulo que no se han descartado, pero este se puede solucionar pasando un filtro de mediana para suavizar los umbrales.

4.2.1.3 Suavizado de umbral

La finalidad de este paso es reducir el valor de los umbrales que no se han podido descartar en el proceso anterior. Lo mas práctico es realizar un

0.6824	0.7569	0.6941	0.6275	0.6667	0.7333	0.8471	0.698	0.8039	0.7294	0.7961
0.7333	0.7843	0.702	0.6275	0.6706	0.6667	0.8588	0.851	0.7216	0.7608	0.798
0.7176	0.749	0.698	0.6471	0.6275	0.6667	0.7725	0.7725	0.7	0.7451	0.9882
0.702	0.7098	0.6667	0.6196	0.6431	0.6667	0	0.8039	0.7392	0.7373	0.7686
0.7137	0.7294	0.7059	0.6431	0.6431	0.651	0.8039	0.8784	0.7804	0.7373	0.7255
0.6863	0.6941	0.6863	0.6353	0.6706	0.7137	0.7373	0	0.7765	0.6843	0.8392
0.7059	0.7098	0.6667	0.6745	0.698	0.6627	0.7216	0	0.7255	0.8471	0.7647
0.6863	0.6941	0.7333	0.6824	0.6549	0.6471	0.6784	0	0.7569	0.7412	0.7275
0.6667	0.702	0.7333	0.6745	0.6667	0.8588	0.698	0.7451	0.8235	0.7294	0.9706
0.6706	0.7216	0.6941	0.6627	0.6627	0.7686	0.7961	0.7608	0.7216	0.7176	0.7588
0.7686	0.651	0.6627	0.6706	0.7216	0.8078	0.8039	0.7608	0.8353	0.7686	0.7804
0.6667	0.6863	0.702	0.7176	0.7765	0.8176	0	0.698	0.7294	0.7922	0.8039
0.702	0.894	0.7608	0.7686	0.7608	0.7686	0.7725	0.6784	0.7922	0.8078	0.8549
0.7176	0.6902	0.6706	0.7176	0.7451	0.7137	0.7294	0.7255	0.7804	0.8118	0.851

Figura 4.17: Umbrales Otsu en CP acotados

filtro estadístico de ordenamiento de mediana.

En este proceso no interactúan los umbrales descartado anteriormente, ya que pueden provocar desplazamientos del valor deseado añadiendo ceros al principio, el proceso final se puede apreciar en la siguiente figura:



Figura 4.18: Resultado final de aplicar el método de Otsu

4.2.2 Detector de borde de Canny

Respecto a los umbrales que usa este método, lo importante es escoger un umbral alto para obtener los bordes principales, pero no muy alto ya que se pierde información importante.

Este umbral es diferente para cada imagen por ser relativo al gradiente, cada imagen tiene sus cambios de color y hace que difiera el módulo del gradiente. La fundamental es buscar un porcentaje del módulo del gradiente muy similar al proceso usado en segmentación por umbral mencionado en 3.2.1. Primera derivada.

Los valores óptimos de umbral alto suelen ser de un 5/10 % y para el bajo entre un 15/25 %. En la siguiente figura se muestra un ejemplo de aplicar Canny a una imagen del proyecto:

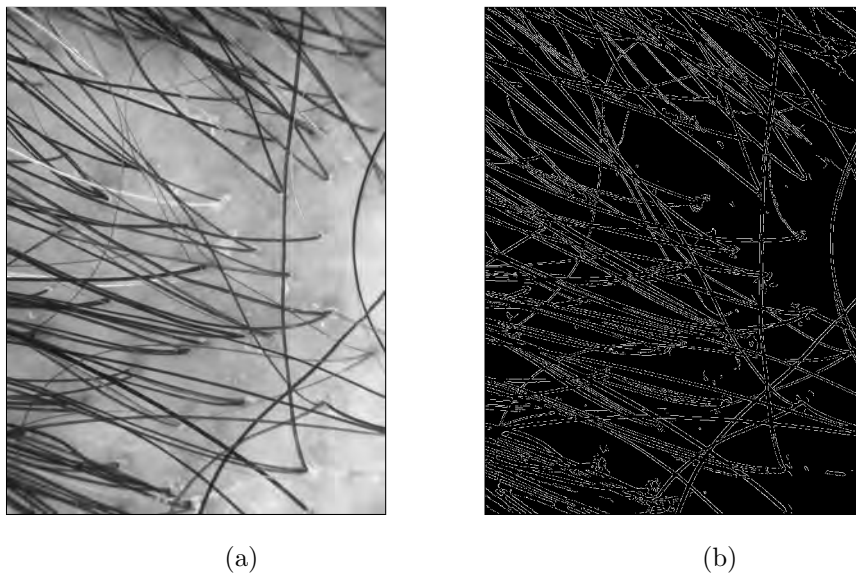


Figura 4.19: (a) Imagen original en CL, (b) Resultado de aplicar Canny

En este momento la segmentación no está completa. Queda rellenar el espacio entre los bordes de los pelos mediante operaciones morfológicas. La dilatación no es una buena opción para rellenar ya que, va a ensanchar los pelos e incluso juntar varios que estén muy próximos y no se va a poder tomar medidas reales. Con lo que primero se debe dilatar y luego erosionar para eliminar el efecto del dilatado indeseado, operación de cierre.

Ahora el problema radica en el tipo de elemento estructurante a usar, el pelo al final se puede ver como líneas en diferentes direcciones; lo más fácil y adecuado es usar un elemento estructurante de tipo línea, e ir rotando este para conseguir el rellenado en todas las direcciones.

Como solución se propone usar un elemento línea de tamaño 12 píxeles, un ancho medio de los pelos, e ir rotando de 15 en 15 grados hasta 180°.

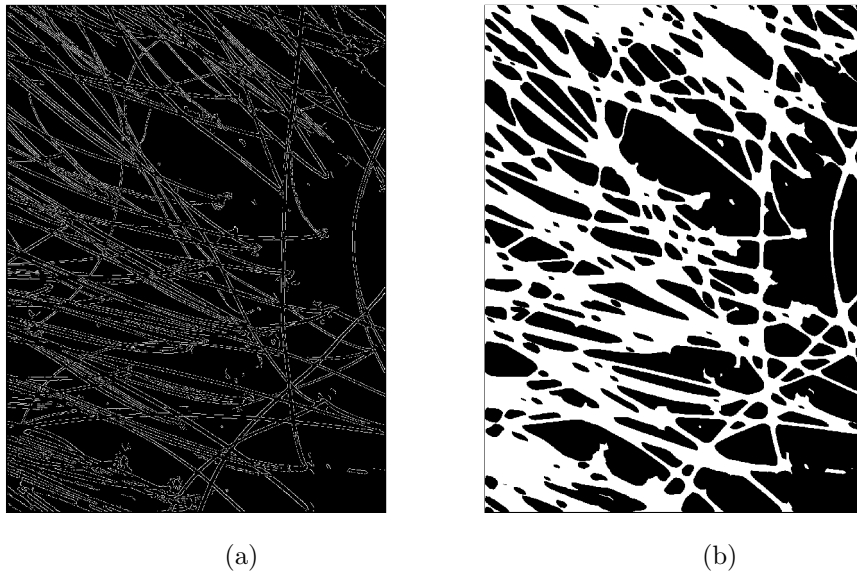


Figura 4.20: (a) Aplicando Canny, (b) Aplicando operación de cierre a la salida de Canny

Lo malo de este proceso que si los pelos estan muy juntos se segmentan como si fuera uno solo.

4.2.3 Derivative of Gaussian (DoG)

En este apartado se hace uso de la explicación del punto 2.2.1.1, para segmentar el cabello usando filtros DoG.

La idea es extender la respuesta del centro del pelo para segmentar todo el pelo, haciendo uso de la formula 2.1 de esta memoria:

$$C_i(x, y) = \begin{cases} R_i(x - d, y) + |R_i(x + d, y)| & \text{si } R_i(x - d, y) > 0 \text{ \& } R_i(x + d, y) < 0 \\ 0 & \text{otros casos} \end{cases}$$

Si analizamos la formula vemos un factor d , que resta o suma en la componente x . Este factor maximiza la función en el centro del pelo ya que si se selecciona d como el ancho medio del pelo se obtendrá en el respuesta el máximo valor posible.

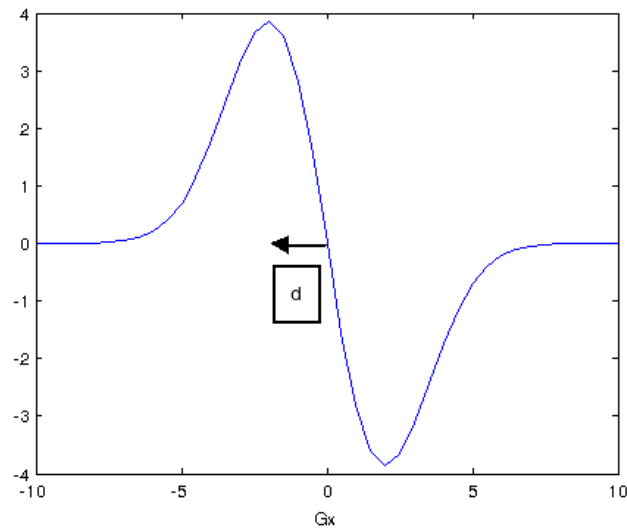


Figura 4.21: Ejemplo de la utilidad del factor d

Lo que se pretende es conseguir una respuesta que a partir del máximo no decrezca, como se muestra en la figura 4.21, si no que siga creciendo hasta llegar al centro, para después decrecer hasta el mínimo y así poder segmentar el cabello. La siguiente figura ilustra la respuesta de ecuación 2.2:

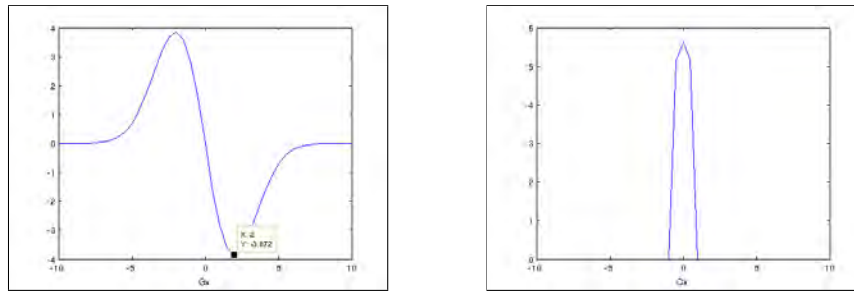


Figura 4.22: Ejemplo de obtención del centro del pelo (a) Ejemplo de filtrado DoG, (b) Resultado de aplicar la obtención del centro a la salida del filtrado DoG

Este ejemplo está hecho en solo una dimensión pero sirve para hacerse una idea del funcionamiento. En imágenes se tienen 2 dimensiones (x,y) y por este hecho se debe buscar la dirección del pelo, filtrando con diferentes direcciones de DoG.

En este proyecto la solución escogida es usar 4 direcciones, vertical, horizontal y las diagonales y cada filtro DoG tiene una de estas direcciones. Un ejemplo de filtros en estas cuatro direcciones está ilustrado en figura 2.4 en el apartado 2.2.1.1 Derivative of Gaussian(DoG).

La máxima respuesta del filtrado será la adecuada a la orientación, el proceso es filtrar por las 4 direcciones, aplicar la formula 2.1 en función de la orientación y ver para cada píxel la máxima respuesta.

Si nos basamos en las imágenes del proyecto, la respuesta no es tan pura como en la figura 4.21, puede aparecer respuesta fuera del cabello provocada por ruido en la imagen o por respuestas de otras orientaciones que generan

ruido. Lo importante es escoger un valor de σ y en función de este, estimar un umbral por el cual se etiqueta cada píxel como pelo o no. Se eligió un valor de σ igual a 3.5 porque en un principio se vió que la respuesta era mas o menos clara, con bajos cambios de valor, y luego después se vió que es un valor que en relación al ancho, ver Anexo A, es un valor mas o menos medio del ancho del cabello en las imagenes del proyecto.

Por último, es necesario estimar un umbral por el cual se considera pelo o no para binarizar la imagen. Este valor es importante en la calidad de la segmentación, debe ser un valor que deje pasar un número pequeño de valores de píxeles de no pelo, pero cercionando de que no se descartan valores de píxeles asociados a pelo.

Pero antes, se debe aplicar una cota al nivel de gris de cada píxel, por el cual descartamos ese píxel si su nivel es mayor de 220. En la siguiente figura se expone el resultado de aplicar este umbral en una de las imágenes del proyecto:

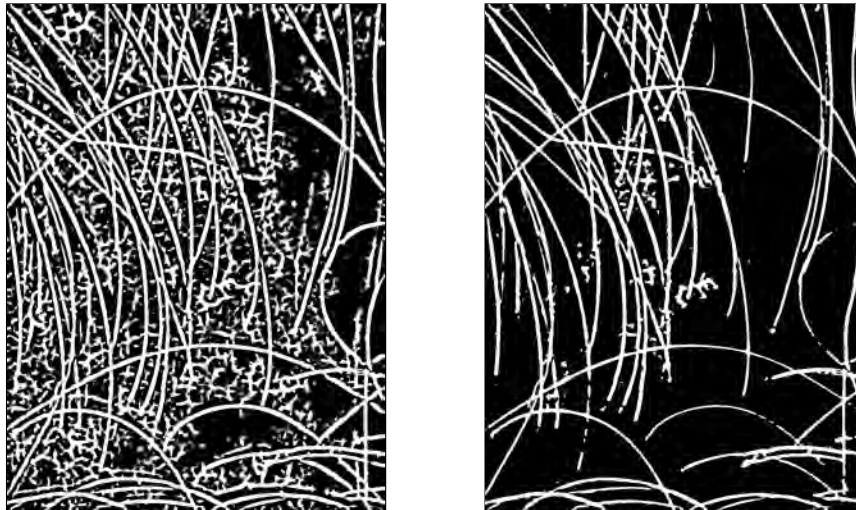


Figura 4.23: Ejemplo de una cota de nivel de gris para cada píxel (a) Antes, (b) Después

Como se puede ver en la figura 4.23, quedan menos píxeles que se deben descartar. Para escoger el valor de la cota de binarización se ha usado el histograma de la respuesta actual:

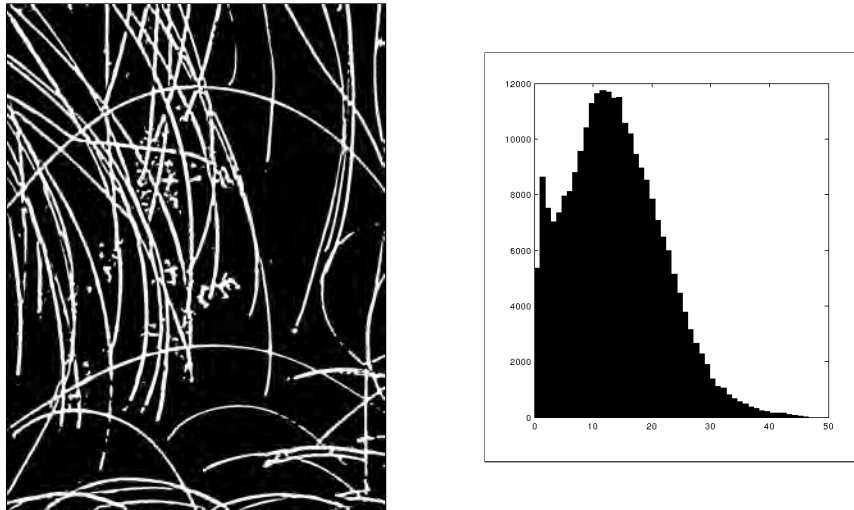


Figura 4.24: Histograma del resultado de aplicar una cota al nivel de gris(a)
Resultado al que pertenece el histograma, (b) Histograma

Se puede apreciar en el histograma un pequeño máximo a la izquierda, que resalta bastante, esos son los píxeles que se quiere descartar. Este máximo se localiza sobre un valor aproximado de 3, se eligió un valor de 3.5 viendo los histogramas de las imágenes de entranamiento del proyecto, ver Anexo B.

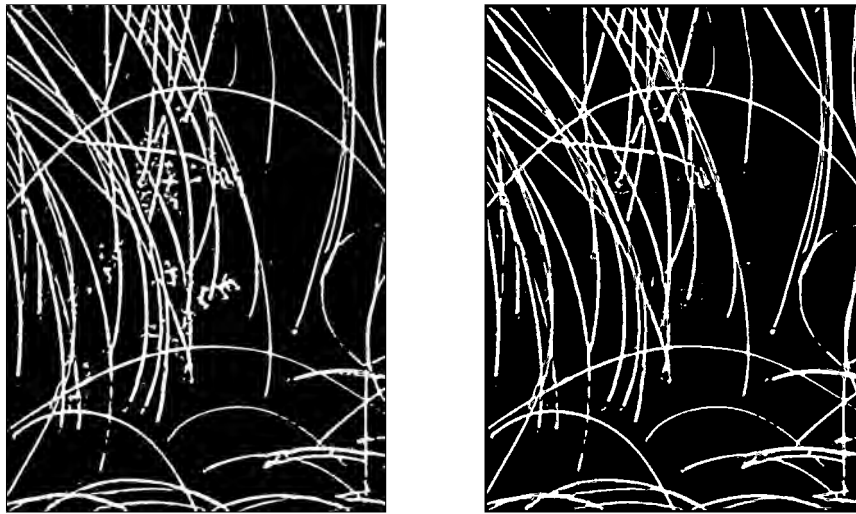


Figura 4.25: Ejemplo de aplicar una cota en la respuesta del centro en segmentación DoG(a) Antes, (b)Despues

Se debe añadir, que en un principio se usaba un valor de d estimado visualmente, no computacionalmente, para realizar las pruebas evaluación. Ese valor se usaba para todos los píxeles de la imagen. Con la estimación del ancho, apartado 4.4.2, se puede usar un valor de d para píxel en función del ancho del pelo a segmentar, mejorando la segmentación.

4.2.4 Borrado de pequeños objetos

En la segmentación es muy probable la aparición de pequeños objetos segmentados, que no son realmente pelo.

Una buena solución es aplicar un criterio por el cual, los objetos seg-

mentados con área menor a un número de píxeles no se consideran objetos reales y han de ser borrados; un procedimiento a usar ser el explicado en el punto 3.5 Apertura de área.

Los valores siguientes son los parámetros introducidos en cada segmentación que se han visto que se adecuan bien a cada método:

Método	λ
Otsu	200
Canny	500
DoG	150

Cuadro 4.1: Valores de λ en apertura de área

Ejemplo:

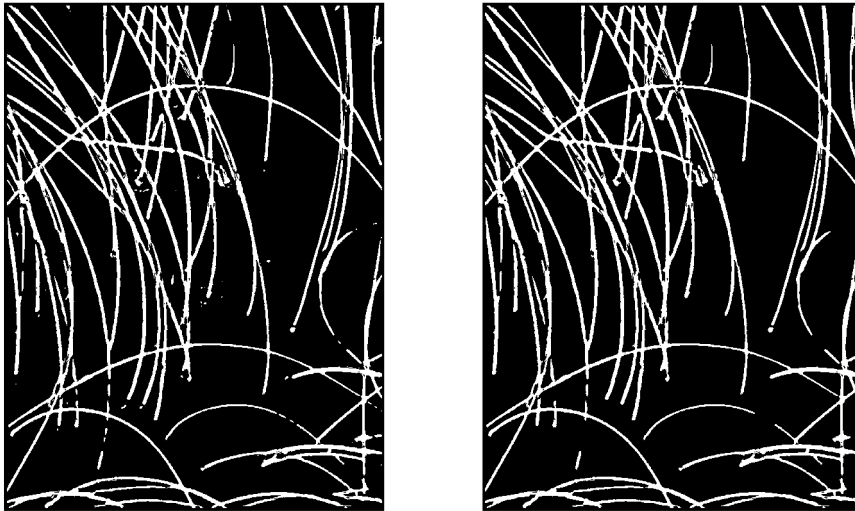


Figura 4.26: Aplicando apertura de área en DoG(a) Antes, (b)Después

4.3 Etiquetado

Es importante después de la segmentación hacer un etiquetado de los objetos, para poder trabajar con ellos en vez de trabajar con toda la segmentación.

Lo que se quiere conseguir con el etiquetado es que cada objeto independiente, no este conectado a otro objeto, tenga una etiqueta identificativa.

El algoritmo usado es el propuesto en [22] que etiqueta objetos en una imagen binaria. El algoritmo en un principio hace distinciones de objetos por filas indistintamente de si este objeto continúa en la fila anterior o superior.

Sigue estos pasos:

- Codificar los “run” de la imagen de entrada: Lo primero es definir los llamados “run”, cada grupo de píxeles en una misma fila es un “run”, puede haber varios “run” en una misma fila al igual que ningún run en la fila. Un “run” está definido por la columna por la que empieza, por la columna que acaba y por su etiqueta preliminar. Al empezar el algoritmo la etiqueta preliminar de los “run” se establece como cero.

Por cada fila se establece el “run” por el que empieza la fila y el “run” por el que acaba.

Teniendo esta imagen se va a establecer los valores de los “run” y la distribución en cada fila:

$$I = \begin{bmatrix} 1 & 1 & 0 & 1 & 1 \\ 1 & 1 & 0 & 0 & 1 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 1 & 1 & 1 \end{bmatrix}$$

	START_COL	END_COL	PR_LABEL
1	1	2	0
2	4	5	0
3	1	2	0
4	5	5	0
5	1	3	0
6	5	5	0
7	2	5	0

Cuadro 4.2: Codificación de los “run”

	ROW_START	ROW_END
1	1	2
2	3	4
3	5	6
4	0	0
5	7	7

Cuadro 4.3: Codificación de las filas en función de los “run”

- Asignación de las etiquetas preliminares y construcción de la clase equivalente; Antes se debe definir una estructura de etiquetas, puede que un grupo de etiquetas preliminares englobe un conjunto de “run” colindantes tenga una etiqueta, es como agrupar etiquetas en una etiqueta diferente. Esta estructura se aplica porque los “run” pueden estar conectados unos con otros, hay equivalencia. Dando lugar a un nuevo concepto la clase equivalente.

La idea es ir recorriendo las filas en función de los “run”, e ir dando

valor a la etiqueta preliminar en función de si tiene un “run” colindante en la fila anterior, si es así se le añadiría su etiqueta preliminar o por el contrario una nuevo valor de etiqueta preliminar.

- Reetiquetado en función de la clase equivalente: Se vuelve a recorrer las filas de la ultima a la primera, para terminar de etiquetar los objetos en función de la clase equivalente.

4.4 Extracción de características

Ya habiendo obtenido la segmentación ahora queda obtener las características de los pelos. En esta sección se expone el diseño para el cálculo de la orientación, ancho y color.

4.4.1 Estimación de la orientación

Para el cálculo de la orientación se basa en la segmentación de Otsu y DoG obteniendo la orientación del centro del pelo por un lado y por otro lado la orientación de los bordes del pelo.

- Centro del pelo: Para la orientación del centro del pelo se hace uso de la segmentación DoG, pero en vez de estar basada en el factor d como ancho medio del pelo es un valor de 2 para obtener solo los cruces por ceros y acotar al centro del pelo.
- Borde de los pelos: Se hace uso del DoG pero obteniendo los valores de la respuesta al cuadrado para obtener el mismo valor en la posición del mínimo, valores altos en la localización en los bordes y los bajos en el centro. Para no obtener ruido en la respuesta se filtra el resultado con la segmentación obtenida en los pasos anteriores.

Ahora solo queda superponer los 2 resultados para obtener la orientación total de los pelos; aparte en cada proceso al final se filtra con un filtro de mediana para suavizar.

Este algoritmo sigue estos pasos para eliminar el máximo ruido posible ya que si se obtienen los bordes extendiendo el centro del pelo se añade ruido al igual que si se quiere obtener el centro mediante el cálculo de los bordes.

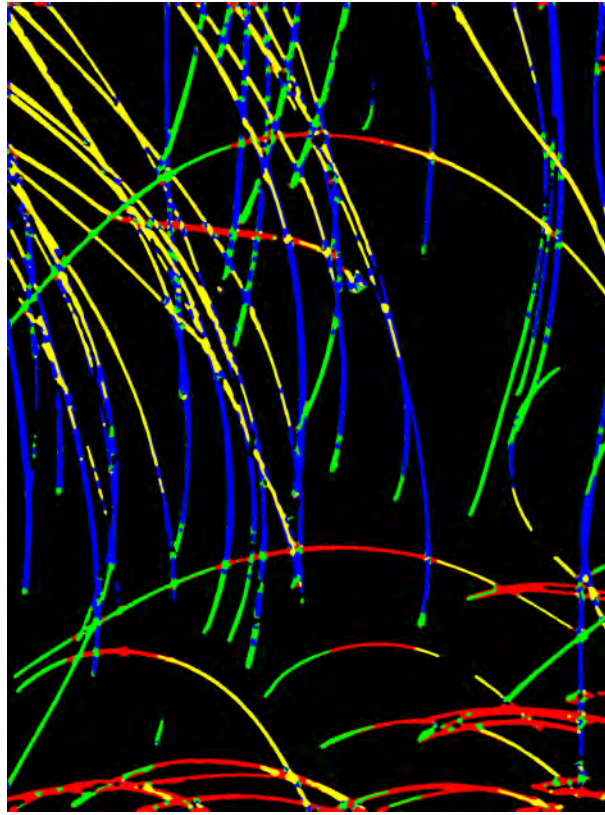


Figura 4.27: Orientación representado por colores. Rojo simboliza 0° , verde 45° , azul 90° y amarillo 135°

4.4.2 Estimación del ancho

Para la estimación del ancho se han usado 2 procedimientos pero ambos están basados en filtros derivativos gaussianos:

4.4.2.1 Basado en filtros DoG normalizados

Este proceso se basa en la utilización de filtros DoG normalizados, para detectar pelos con diferentes anchos; para ello se debe realizar una normalización del filtro, hacerlo invariante, añadiendo un factor σ

$$DoG_N = \sigma DoG$$

El procedimiento se basa en obtener previamente una relación de la σ del filtro con el ancho del pelo. Lo importante es entender que con la σ adecuada al ancho del pelo, el filtrado dará una respuesta máxima en comparación con las diferentes respuestas con diferentes σ .

Lo fundamental es buscar una relación del ancho del pelo con σ , y para ello se usaron imágenes sintéticas en las cuales se muestra una línea vertical con un ancho diferente. Se usaron un total de 27 imágenes, donde la primera imagen tiene un ancho de 3 píxeles y la última un ancho de 30 píxeles, ver Anexo B.

Cada imagen se filtra con un rango de valores de σ , obteniendo tantas imágenes filtradas como valores y obtener la imagen de máximos, cada valor del píxel corresponde al valor máximo de todas las imágenes filtradas. Después, se obtiene cual es la imagen filtrada que cumple la existencia de más número de máximos en la imagen de máximos y dará la σ adecuada a este ancho.

En el anexo A se expone la relación de la σ con el ancho medido en píxeles. Para la estimación del ancho, se usarán todos los valores expuestos y con cada valor se filtrará con cuatro orientaciones diferentes 0° , 45° , 90° y 135° .

La idea es ir recorriendo todos los valores de σ , en cada valor se realiza tantos filtrados como orientaciones diferentes con lo que se obtiene la imagen de máximos, para esa sigma en función de la orientación y así para todas las σ , guardando todas las imágenes de máximos en una variable. Ahora se tienen tantas imágenes de máximos como σ . La finalidad es conseguir la imagen de máximos que obtenga en cada píxel el máximo de todas las imágenes de máximos calculado previamente, es saber en cada píxel cual es el máximo de todas las σ .

Por ultimo se ve cuantos máximos corresponden a cada σ , obteniendo la relación inmediata en píxeles de anchura. Para mostrar el resultado final se almacena en variables el numero de máximos, para poder mostrar un histograma en función del ancho en píxeles, dando un histograma global.

La siguiente figura es un ejemplo del cálculo de anchos para este método:

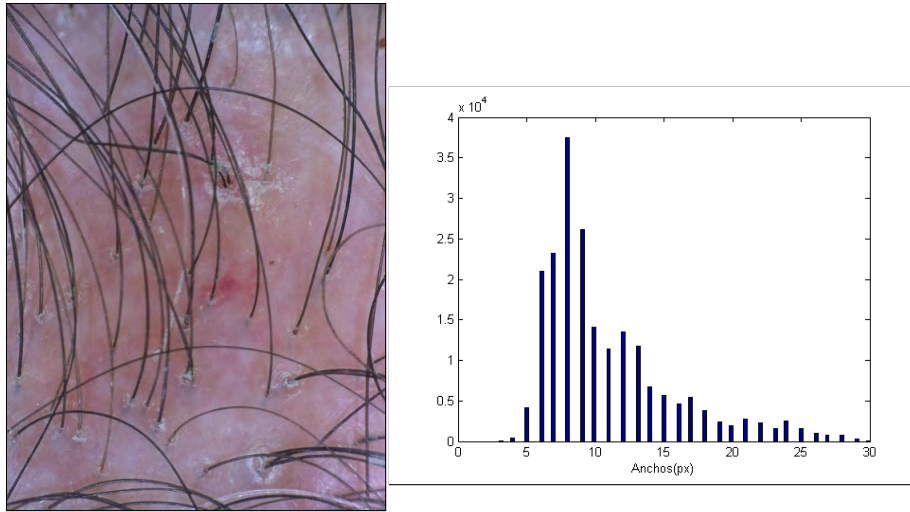


Figura 4.28: (a) Imagen original, (b) Estimación del ancho basado en DoG normalizados

4.4.2.2 Basado en la respuesta del filtro DoG

La idea fundamental aquí, es diseñar una máquina de estados que haga seguimiento de la respuesta del filtro DoG, mencionado previamente en el punto 3.2.1.1 DoG. Esta respuesta está definida con un máximo al principio del pelo y un mínimo al final o viceversa, en función de la orientación, teniendo una distancia d de los máximos/mínimos al centro del pelo.

Los estados definidos son:

- Estado 0: Se entra en la respuesta derivativa guassiana.

- Estado 1: Se crece hasta llegar al máximo
- Estado 2: Se decrece en valores positivos, se guarda el punto al entrar en el estado, ya que es donde empieza el pelo.
- Estado 3: Se decrece en valores negativos hasta llegar al mínimo.
- Estado 4: Se crece hasta un valor proximo a cero.
- Estado 5: Se deja de crecer, se guardan los valores de máximo y mínimo.

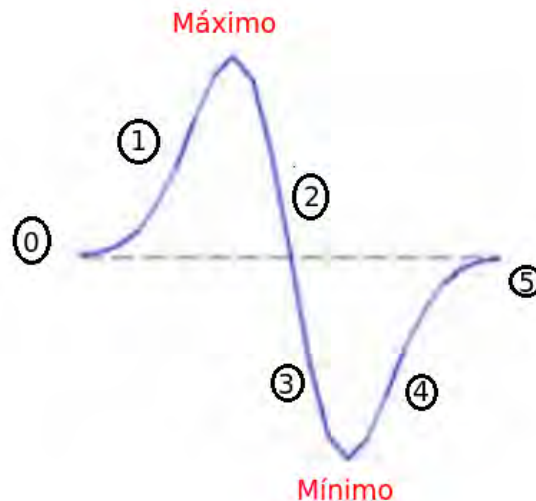


Figura 4.29: Representación de la máquina de estados

Si se cumplen todos los estados, indica la presencia de un pelo y para calcular su ancho no es más que hacer una resta entre punto mínimo y máximo. Lo importante es diseñar la máquina para poder trabajar con 4 direcciones, 0° , 45° , 90° y 135° . y poder estimar un valor real del ancho del pelo en función de la orientación.

Lo importante es seleccionar una σ del filtro, que permita tener una respuesta continua para poder seguir bien los estados, se eligió una σ igual

a 3 que si se ve su relación con el ancho, Anexo A, suele ser un valor cercano al ancho medio de los pelos en las imágenes.

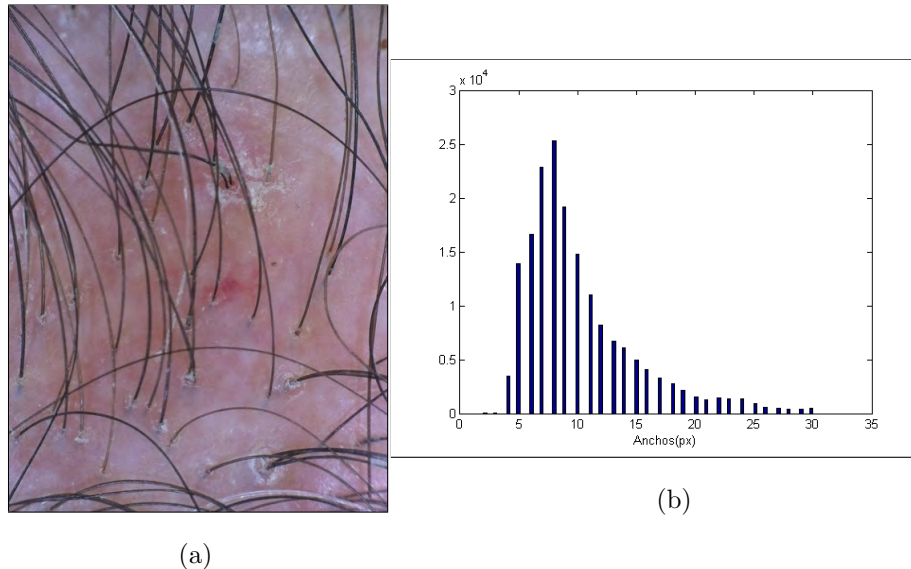


Figura 4.30: (a) Imagen original, (b) Estimación del ancho basado en la respuesta del filtro DoG y máquina de estados

4.4.3 Estimación del color

En este apartado el propósito principal, es conseguir los colores predominantes en los pelos obtenidos mediante la segmentación, y tener una idea de la evolución del color del cabello durante el tratamiento.

Primero se exponen los espacios de colores y sus características:

4.4.3.1 Espacios de color

Los espacios de color[8] son modelos para representar el color, muchos de ellos están formulados en función de la respuesta del ojo humano, XYZ, y otros para un rápido proceso maquina, RGB. Hay que minimizar el número de dimensiones y maximizar el número de colores, la mayoría de los espacios

de colores tienen 3, dimensiones, son suficientes para describir un conjunto amplio de colores(gamut).

- RGB: El espacio de color de RGB está definido por tres colores, rojo(R), verde(G) y azul(B). Los colores se representan por la mezcla aditiva de éstos y forma un cubo donde cada eje pertenece a un color:

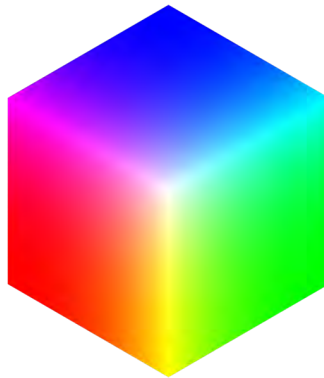


Figura 4.31: Cubo RGB

Lo más habitual es utilizar un byte, uint8, para representar cada dimensión, permitiendo un total de 16777216 colores. Este espacio de color es usado por defecto por matlab para representar el color.

- HSV: En este espacio de color tenemos 3 dimensiones [9]
 - Hue: Es lo que identifica a cada color, el matiz, se representa como un círculo los colores primarios están cada 120° , al recorrer este círculo se irán recorriendo todos los colores.
 - Saturation: Representa la saturación del color, es una medida de la viveza del color se mide en un rango de 0 % a 100 %. En medida en que nos vamos acercando a 0 el color se va perdiendo hasta llegar al blanco o gris.

- Brillo: Comprende un rango entre 0 % y 100 %, donde el 0 % es el negro y el 100 % es el blanco.

Si nos fijamos en la siguiente imagen, podemos ver cómo el color va cambiando en medida que va aumentando el ángulo, la saturación es el radio y el brillo es la altura.

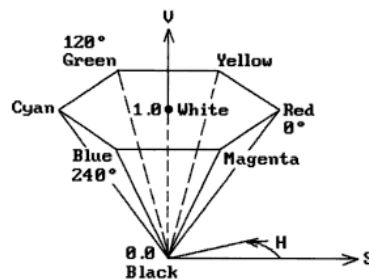


Figura 4.32: HSV

- CIELAB: Es un espacio derivado de CIE XYZ[10], que fue creado por el Comisión internacional de la uliminación (CIE) y se basa en la respuesta del sistema visual humano. Componentes:
 - La componente L determina la luminosidad del color, si L es 0 significa un color negro y 100 color blanco.
 - Para valores positivos de la componente a indican un color rojo y para negativos un color verde.
 - De la componente b se obtienen para valores positivos el color amarillo y en valores negativos el azul.

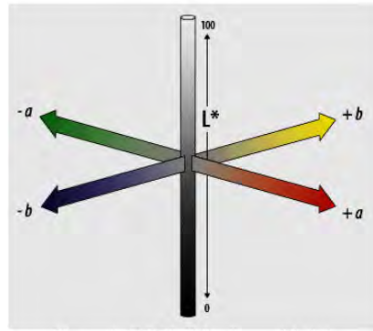


Figura 4.33: Representación CIELAB

4.4.4 Descripción del color

Ya habiendo visto los diferentes espacio de colores, en este proyecto se va a hacer uso del espacio de colores HSV ya que es más fácil manejar los colores gracias a la componente de HUE. Para ello se hace un histograma en función del HUE para ordenar los colores, y dar información visual de los colores predominantes en la imagen.

Cada barra del histograma representa un rango de valores de HUE, y el color que representa cada barra será el color medio del rango. Es preferible un número considerable de barras, para que al representar el color medio sea lo mas preciso posible. Cada barra está definida por el color medio(HUE, saturación y brillo) y número de píxeles que tienen ese color.

Aparte también se dan valores medios de saturación y su desviación típica.

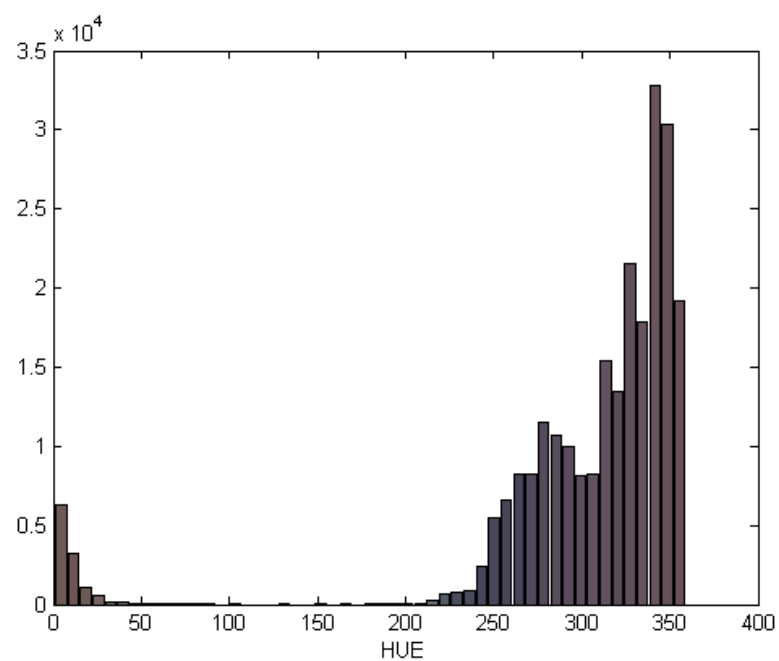


Figura 4.34: Histograma en función de HUE

Capítulo 5

Evaluación

En este capítulo se van a evaluar las diferentes alternativas de diseño que han sido propuestas en el capítulo anterior.

Se presentarán las ventajas y desventajas de cada algoritmo y se determinará cual es el mejor algoritmo en base a las medidas de evaluación.

5.1 Máscaras de referencia

Para evaluar la calidad de la segmentación de los diferentes métodos, se hace uso de máscara de referencia o en ingles “Ground truth”, la cual este hecha en función de la imagenes del proyecto con un programa de segmentación manual, “segmentation tool”.

El propósito es conseguir máscaras binarias que indiquen la presencia de pelo, etiquetando con un uno y los demás con un cero. No debe haber píxeles que en imagen real son pelos y estos píxeles estan etiquetados como ceros en la mascara de referencia.

5.2 Medidas de evaluación

Hay que evaluar los metodos de segmentación y los metodos de estimación del ancho, para ello se utilizan 2 medidas:

- “Hammoude Distance (HD)”: Es una medida [18] de estimación de la calidad de la segmentación que usa la máscara de referencia y la máscara de la segmentación para obtener un resultado. Se define como:

$$HD = \frac{num(GT \cup SM) - num(GT \cap SM)}{num(GT \cup SM)}$$

Siendo GT las máscara de referencia, SM la máscara de segmentación, \cup union, \cap intersección y num es el número de píxeles.

El resultante de la unión de dos matrices, da un valor de uno en la posición donde haya un uno en cualquiera de las dos matrices, por otro lado la intersección da como resultado un un si en las dos matrices coinciden con el valor de uno, en caso contrario es cero.

Si se analiza la formula y la segmentación es perfecta este valor de evaluación daría un cero, ya que la unión es igual que la intersección. La segmentación es mejor en medida en que HD se acerque a cero.

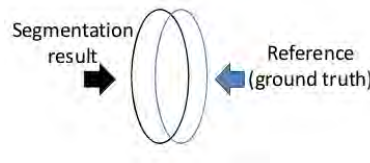


Figura 5.1: Representación HD

- Precisión del ancho: Esta parámetro medido en porcentaje(%) evalúa la precisión de la estimación del ancho. La precisión se evalúa de esta forma, si se sabe que un objeto tiene ancho 8 y se ha estimado un ancho 7.9 su precisión es $7.9/8 = 98,75\%$.

5.3 Resultados

Aquí se muestran los resultados de evaluar los métodos de segmentación, con las 2 componentes usadas durante el proyecto. El método de Otsu no aplica en CL porque se descartó en la segmentación. DullRazor funciona a parte del proyecto y se ha evaluado para tener una referencia.

Lo que se muestra en el siguiente cuadro, son los valores medios de aplicar la medida de evaluación “Hamoude Distance” en las imágenes del proyecto, marcando el valor en verde para la mejor evaluación y en rojo para la peor evaluación:

Método	HD(CP)	HD(CL)
Otsu	0.348	NA
Canny	0.482	0.515
DoG	0.43	0.445
DullRazor	0.502	0.502

Cuadro 5.1: Evaluación de la segmentación

Como se muestra en verde, Otsu con la componente principal da mejores resultados que las demás y es la que se eligió para usar durante parte del proyecto

Para la evaluación de la estimación del ancho se ha usado una imagen sintética, ver Anexo D, con diferentes líneas y diferentes anchos. Para evaluar la estimación se tiene en cuenta la precisión del ancho y la HD.

Se debe incluir que los métodos de estimación del ancho estiman el ancho del vello en cada píxel, dando como resultado una matriz de anchos. Otra buena forma de medir la calidad es evaluando la matriz binaria obtenida de la matriz de anchos; un valor diferente a 0 en la matriz de anchos da como resultado un valor de 1 en la matriz binaria en ese píxel, en caso contrario un 0. La calidad de la estimación se mide en función de los píxeles reales de pelo que ha estimado, HD, y de la precisión del valor del ancho.

El siguiente cuadro, muestra la evaluación de los estimadores de ancho aplicando la imagen sintética:

Método	HD	Precision
DoG normalizados	0.079	94,95 %
Respuesta DoG	0.085	90,94 %

Cuadro 5.2: Evaluación de la estimación del ancho

Si se compara la medida HD de los estimadores de anchos, con los métodos de segmentación el valor es bastante menor en los estimadores de anchos. Esto se debe a que la imagen usada es en terminos de segmentación mas facil de segmentar, son líneas negras sin ruido. Si se aplicará los metodos de segmentación sobre esta imagen sintética se obtendría un valor aún más bajo.

Se puede ver que el uso de filtros DoG normalizados da mejores resultados y es la solución elegida para la implementación del código final.

Capítulo 6

Planificación y presupuesto del proyecto

En este capítulo se describe el coste y tiempo del proyecto así como su planificación.

6.1 Planificación

El proyecto se ha dividido en estas fases:

- Fase 1: Planteamiento del problema: Inicio del proyecto, toma de contacto, estudio de problema, alternativas y asimilación de conceptos.
- Fase 2: Diseño del sistema: Habiendo estudiado las alternativas y las características de cada una, se procede a codificar el diseño técnico de segmentación y extracción de características del pelo.
- Fase 3: Pruebas: Evaluar la calidad del diseño técnico para escoger la mejor alternativa.
- Fase 4: Documentación: Fase para la elaboración de la memoria.

La siguiente figura representa un diagrama Gantt [21] en el que están definidas las fases del proyecto y el tiempo para cada fase. Da una idea de la dificultad de las fases del proyecto, e incluso puede verse cómo hay fases que se solapan para poder optimizar el tiempo hasta la finalización del proyecto.

CAPÍTULO 6. PLANIFICACIÓN Y PRESUPUESTO DEL PROYECTO 83

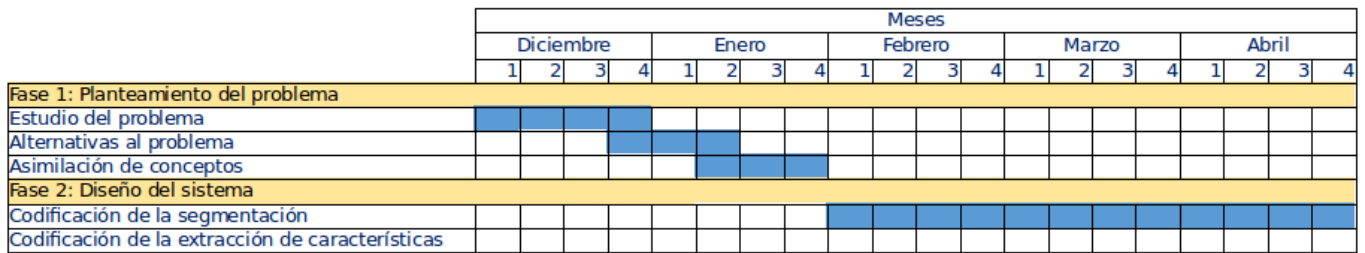


Figura 6.1

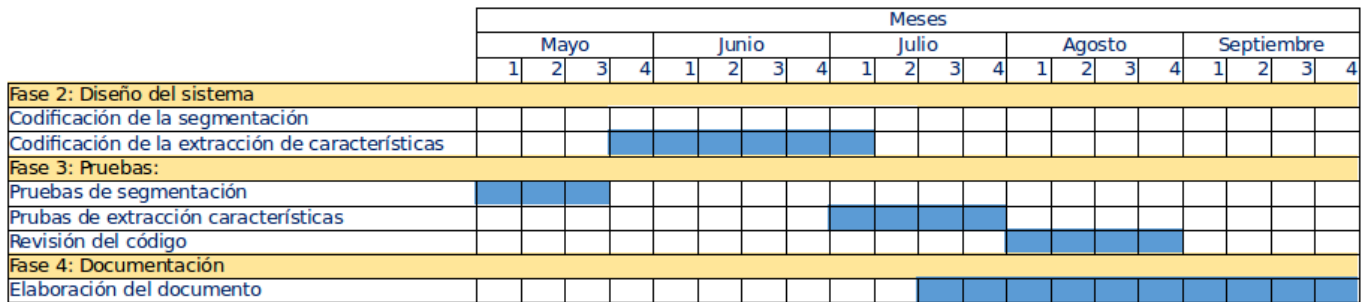


Figura 6.2

Figura 6.3: (a) Parte 1 del Gantt, (b) Parte 2 del Gantt

6.2 Presupuesto

El presupuesto se puede dividir en costes. Materiales y personales.

- Materiales:

- Espacio de trabajo: El espacio de trabajo ha sido un pequeño local con precio mensual de este estimado en 400€ al mes incluidos gastos de luz, calefacción e internet; si el proyecto tuvo una duración de 10 meses hace un coste total de 4.000€.
- Ordenador personal: El ordenador personal usado es un Asus con sistema operativo linux, con un precio de 600€ y suponiendo una vida útil de un portátil de seis años, obtenemos un coste de 100€

al proyecto.

- Software: Para la realización del proyecto se usó el software Matlab R2012 A para ubuntu, versión de estudiante con un coste de 315€.

Tipo	Coste
Espacio de trabajo	4.000€
Ordenador personal	100€
Software	315€
Total	4415€

Cuadro 6.1: Costes materiales

- Personales: Se ha estimado las horas de dedicación del alumno y del tutor. Asumiendo veintidós días hábiles por mes y 4 horas de dedicación al día obtenemos:

Persona	Horas	Honorarios	Total
Ingeniero	880	30€/hora	26.400€
Director de proyectos	100	60€/hora	6.000€
Total			32.400€

Cuadro 6.2: Costes personales

- Total: Teniendo en cuenta todos los costes:

Descripción	Coste
Costes materiales	4415€
Costes personal	32.400€
Subtotal	36.865 €
IVA(21 %)	7751,65 €
Total	44.556,65€

Cuadro 6.3: Coste total

Chapter 7

Conclusion and futures lines

7.1 Conclusions

Many diferents alternatives have been described during the reports, explaining all proterties of each one and choosing the best.

The problem have been studied with a detailed analysis, but the images shortage in the project has not been done test images, only train images.

But still the results are realiable and no future problems are expected in different images.

In regards to implement this viability study to a comercial product, we should increase the images in database. The most important is to analyzes images in a different moment of the treatment and verifies system functionality, knowing a priori the patient's condition has improved or deteriorated and analyzes the images to test the system's results are equal.

7.2 Futures lines

We can consider:

1. Perfection system: Increasing the number of train images and getting test images.
2. Label enhancement: Labelled does not distinguish the orientation of hair. It is important distinguish the intersection hair and does not label the diferent hair like one, label as a function of orientation to distinguish the different hair.

3. Segmentation of light hair: The project has based on segmentation of dark hair and we can not test the system with light hair.
4. Interpolation: An other line is to carry out an interpolation of segmented hair.
5. Creating a graphical interface in order to control the parameter of the system.
6. Develop a comercial software for the diagnostic of alopecia: patient's authorizations, more pictures in database to make the system stronger, fundraising and customers search.

Anexos

Anexo A

Relación σ con ancho

σ	Anchos(px)
1	3
1.25	4
1.5	5
2	6
2.25	7
2.5	8
3	9
3.25	11
3.5	12
3.75	13
4.25	14
4.5	15
4.75	16
5	17
5.25	18
5.75	19
6	20

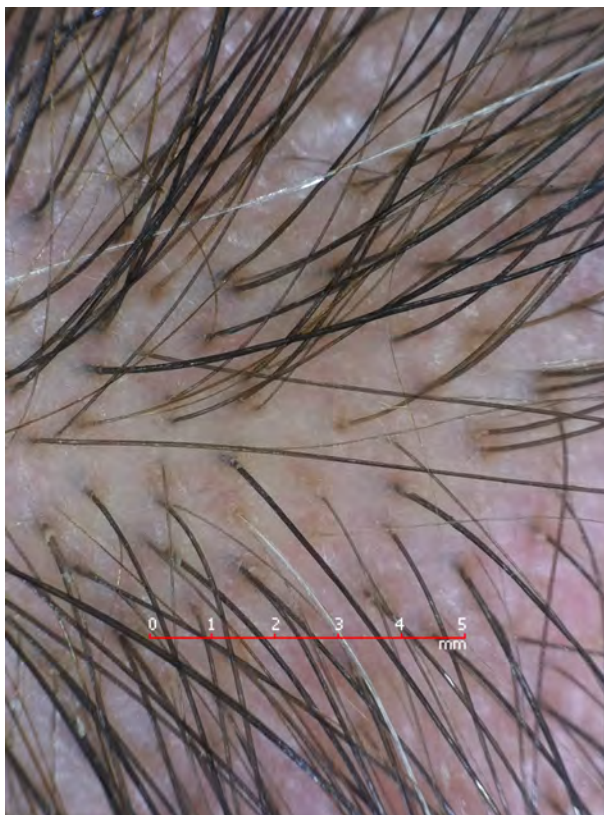
6.25	21
6.5	22
7	22
7.25	23
7.5	24
8	25
8.25	26
8.5	27
8.75	28
9	29
9.25	30

Anexo B

Imágenes usadas en el proyecto











Anexo C

Imágenes usadas para la relación de σ con el ancho en px

Las imágenes siguientes aunque no se aprecia por haber un fondo blanco son imágenes de tamaño 300x300 con una línea en vertical en negro, simulando un pelo. La primera imagen tiene una línea vertical de ancho 3 píxeles y en medida en que se van avanzando las imágenes se va aumentando un píxel el ancho de cada línea hasta llegar a la última imagen con un ancho de 30 píxeles.



Figura C.1: Línea con ancho 3 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX97



Figura C.2: Línea con ancho 4 px



Figura C.3: Línea con ancho 5 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX98



Figura C.4: Línea con ancho 6 px



Figura C.5: Línea con ancho 7 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX99



Figura C.6: Línea con ancho 8 px



Figura C.7: Línea con ancho 9 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX100



Figura C.8: Línea con ancho 10 px



Figura C.9: Línea con ancho 11 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX101



Figura C.10: Línea con ancho 12 px



Figura C.11: Línea con ancho 13 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX102



Figura C.12: Línea con ancho 14 px



Figura C.13: Línea con ancho 15 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX103



Figura C.14: Línea con ancho 16 px



Figura C.15: Línea con ancho 17 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX104



Figura C.16: Línea con ancho 18 px



Figura C.17: Línea con ancho 19 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX105



Figura C.18: Línea con ancho 20 px



Figura C.19: Línea con ancho 21 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX106



Figura C.20: Línea con ancho 22 px



Figura C.21: Línea con ancho 23 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX107



Figura C.22: Línea con ancho 24 px



Figura C.23: Línea con ancho 25 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX108



Figura C.24: Línea con ancho 26 px



Figura C.25: Línea con ancho 27 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX109



Figura C.26: Línea con ancho 28 px



Figura C.27: Línea con ancho 29 px

ANEXO C. IMÁGENES USADAS PARA LA RELACIÓN DE σ CON EL ANCHO EN PX110



Figura C.28: Línea con ancho 30 px

Anexo D

Imagen sintética para la evaluación de la estimación del ancho

La figura de abajo muestra la imagen sintética construida para la evaluación de la estimación del ancho. En ella se se muestra líneas con diferentes anchos y direcciones:

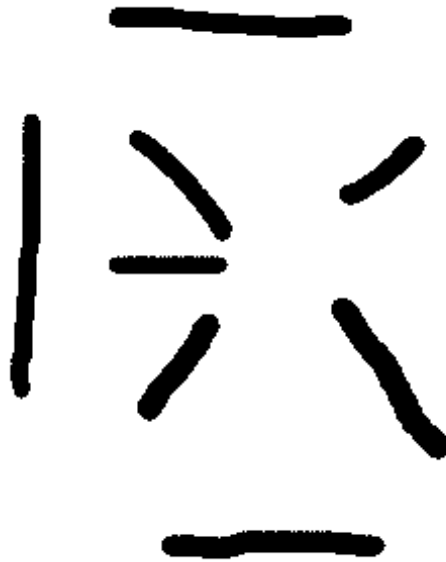


Figura D.1: Líneas con diferentes anchos y direcciones

Bibliografía

- [1] <http://www.tnrelaciones.com/alopecia/> (último acceso 3/09/2015)
- [2] http://chealth.canoe.com/channel_condition_info_details.asp?disease_id=3&channel_id=8&relation_id=55661 (último acceso 6/09/2015)
- [3] Lee T, Ng V, Gallagher R, Coldman A, McLean D. DullRazor: A software approach to hair removal from images. *Computers in Biology and Medicine* 1997;27:533-543.
- [4] <http://trichoscan.com/pages/english/patient-info/what-is-trichoscan.php>
- [5] <http://www.beautyderm.es/estudios-capilares-trichoscan.php?pr=4&sec=7> (último acceso 10/09/2015)
- [6] R. Hoffmann, H. Lüdtke, M. Hoffman-Wecker, B. J. Hughes-Formella: Digital imaging for measuring of hair growth on the human scalp
- [7] Nobuyuki Otsu :Threshold Selection Method from Gray-Level Histogram *IEEE Transactions on Systems, Man, and Cybernetics*, Vol. 9, No. 1, 1979, pp. 62-66.
- [8] Danny Pascale: A Review of RGB Color Spaces ...from xyY to R'G'B' ©2002-2003
- [9] Agoston, Max K: (2005)Computer Graphics and Geometric Modeling: Implementation and Algorithms. London: Springer. pp. 300–306.
- [10] Noor A. Ibraheem, 2 Mokhtar M. Hasan, 3 Rafiqul Z. Khan, 4 Pramod K. Mishra: Understanding Color Models: A Review

- [11] Canny, J.,: A Computational Approach To Edge Detection, IEEE Trans. Pattern Analysis and Machine Intelligence, 8(6):679–698, 1986.
- [12] Edward R. Dougherty :An Introduction to Nonlinear Image Processing
- [13] Luc Vincent :Grayscale area openings and closings, their efficient implementation and applications
- [14] <http://es.mathworks.com/help/images/ref/bwareaopen.html>
(último acceso 11/09/2015)
- [15] Q.L.L. Zhang, J. You, D. Zhang, P. Bhattacharya: Dark line detection with line width extraction, in: Proceedings of the IEEE International Conference on Image Processing (2008) 621–624.
- [16] Freeman, W. T. and Adelson, E. H. (1991): The design and use of steerable filters. IEEE Transactions on Pattern analysis and machine intelligence 13 (9). pp. 891–893.
- [17] Mikolajczyk, K. and Schmid, C: Indexing based on scale invariant interest points, Eighth IEEE International Conference on Computer Vision, 2001. ICCV 2001. Proceedings. Vol. 1, pp. 525 - 531.
- [18] Hair removal methods: A comparative study for dermoscopy images
- [19] Hair removal methods: A comparative study for dermoscopy images
- [20] COMP 558 lecture 11 Oct. 8, 2009 (modified Oct. 10)
- [21] GANTT, H.L. Work, Wages, and Profits. BiblioBazaar, 2010 ISBN 9781171595182.
- [22] Haralick, Robert M., and Linda G. Shapiro, Computer and Robot Vision, Volume I, Addison-Wesley, 1992, pp. 40-48. ISBN 0201108771

- [23] <https://en.wikipedia.org/wiki/Retina> (último acceso 11/09/2015)
- [24] Sahinaz Safari Sanjani, Jean-Baptiste Boin, Karianne Bergen: Blood Vessel Segmentation in Retinal Fundus Images
- [25] Nhi Hoang Nguyen: A hybrid approach to segment hair in dermatoscopic images using a universal kernel.
- [26] Hengameh Mirzaalian, Tim K. Lee, and Ghassan Hamarneh: Hair Enhancement in Dermoscopic Images Using Dual-Channel Quaternion Tubularness Filters and MRF-Based Multilabel Optimization
- [27] Ana Afonso and Margarida Silveira: Hair detection in dermoscopic images using Percolation
- [28] <http://sergiovano.blogspot.com.es/2012/03/tricoscopia-en-el-diagnostico-de-las.html> (último acceso 6/09/2015)
- [29] <http://www.fotofinder-systems.com/index.php?id=216> (último acceso 8/09/2015)
- [30] https://www.boe.es/diario_boe/txt.php?id=BOE-A-1999-23750 (último acceso 12/09/2015)
- [31] Gonzalez, R. C., R. E. Woods, and S. L. Eddins, Digital Image Processing Using MATLAB, Gatesmark Publishing, 2009.